

Taneli Heikkinen

Opasrobotti Tiedekeskus Heurekassa

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Automaatiotekniikka

Insinöörityö

25.11.2013

| | |
|--|--|
| Tekijä(t) Otsikko | Taneli Heikkinen Opasrobotti Tiedekeskus Heurekassa |
| Sivumäärä Aika | 36 sivua + 1 liite 25.11.2013 |
| Tutkinto | insinööri (AMK) |
| Koulutusohjelma | Automaatiotekniikka |
| Suuntautumisvaihtoehto | Prosessiautomaatio |
| Ohjaaja(t) | Koulutuspäällikkö Hannu Hassinen Lehtori Timo Tuominen |
| <p>Tässä insinöörityössä eritellään Heurekan opasrobotin kehitystyön aikana tehtyä suunnittelu- ja tutkimustyötä sekä päätöksiä.</p> <p>Tiedekeskus Heureka suunnittelee ja valmistelee näyttelyä tulevaisuuden kaupungeista ja teknologiasta. Näyttely avautuu vierailijoille kesällä 2014. Osaksi näyttelyä suunnitellaan robottia, joka toimisi näyttelyssä oppaana. Sen tehtävänä olisi johdattaa ihmisiä ennalta määriteltuihin paikkoihin näyttelyssä. Robottijärjestelmä olisi hyvä esimerkki ihmisen ja koneen välisten rajapintojen kehittämisestä ja robotiikasta.</p> <p>Insinöörityössä esitellään opasrobotin prototyyppi ja sen pohjalta rakennettavan opasrobottijärjestelmän ominaisuuksia. Järjestelmän toiminnallisuudet tulevat olemaan samankaltaiset molemmissa kehityksen vaiheissa. Projekti päätettiin toteuttaa avoimen lähdekoodin Festo Robotino -robotilla.</p> <p>Robotin paikoituksen ja navigoinnin perusteet ja toimiva kommunikointi robotin ohjausyksikön ja käyttäjärajapinnan välillä ovat suurimmat haasteet projektissa. Robotin pääasiallinen tehtävä on olla sosiaalinen ja positiivinen ihmisiä kohtaan, joten ulkonäkö, toiminta ja kokonaisvaltainen turvallisuus ovat myös tärkeitä projektin osa-alueita. Robotin toiminnallisuuksien tulisi kiehtoa vierailijoita, ja saada heidät jopa uskomaan, että he ovat oikeasti tulevaisuudessa.</p> <p>Kehitysalustan valinnan lisäksi projektin aikana oli tärkeää tutkia, mikä oli työajan ja budjetin resurssien puitteissa mahdollista. Tässä työssä keskityttiin etsimään tietoa aikaisemmista samankaltaisista sovelluksista, rakentaa järjestelmän osa-alueet ja sovittaa ne yhteen projektin resurssit mielessä pitäen.</p> | |
| Avainsanat | Mobiilirobotti, Opas, Festo, Robotino, Heureka, Navigointi |

| | |
|--|--|
| Author(s) Title | Taneli Heikkinen Tour Guide Robot in Heureka Science Center |
| Number of Pages Date | 36 pages + 1 appendix 25 November 2013 |
| Degree | Bachelor's degree (UAS) |
| Degree Programme | Automation Engineering |
| Specialisation option | Process Automation |
| Instructor(s) | Hannu Hassinen, Didactic Manager Timo Tuominen, Lecturer |
| <p>This thesis describes the details regarding the planning, research and decisions made during the prototype build of the Heureka tour guide robot system.</p> <p>Science Center Heureka is planning on to set up an exhibition about future cities and future technology opening in summer 2014. As a part of the exhibition a robot would function as a tour guide showing the way for the visitors to specified places in the exhibition area. The robot system would be a good example of development of human-machine-interfaces and robotics.</p> <p>A prototype of the robot system and some functions of the robot system to be built are presented in this thesis. The functionalities will be similar both in the prototype and the final robot system. It was decided that the basis for the robot development will be Festo Robotino, the Festo open source mobile robot.</p> <p>The basics for robot orientation and navigation and a functioning communication interface are the main challenges in this project. Because of the robot's main function of being social and positive towards human beings, also the look, behavior and comprehensive safety of the robot are important issues. All of the functionalities of the robot should be implemented so that the science center visitor would truly wonder at the science behind the functionalities of the robot and at its best, would make the visitor think that technology made today actually is the future.</p> <p>It was also necessary to find out what was possible within the given resources program wise or physically and what was not. The focus and the goal in this project was to collect knowledge of completed and working applications, build the necessary functions and fit the pieces into each other.</p> | |
| Keywords | Mobile robot, Tour guide, Festo, Robotino, Heureka, Navigation |

Sisällys

Lyhenteet

| | | |
|-------|--|----|
| 1 | Johdanto | 1 |
| 1.1 | Projektin osapuolet | 2 |
| 1.1.1 | Festo Oy | 2 |
| 1.1.2 | Tiedekeskus Heureka | 2 |
| 2 | Opasrobotin viitekehys | 3 |
| 2.1.1 | Robotin stereotyyppi | 3 |
| 2.1.2 | Kehitysympäristö yhteiskunnallisen robotiikan alalla | 4 |
| 2.2 | Aikaisemmat opasrobotinsovellukset | 5 |
| 2.2.1 | Rhino ja Minerva -robotit | 5 |
| 2.2.2 | Urbano -robotti | 7 |
| 2.2.3 | Toyota TPR Robina -robotti | 8 |
| 3 | Navigointi ja paikoitus | 8 |
| 3.1 | Paikoitus | 9 |
| 3.2 | Navigointimallit | 10 |
| 3.2.1 | Suunnitelmallinen navigointi | 10 |
| 3.2.2 | Reaktiiviset navigointimallit | 10 |
| 3.2.3 | Robotin asemasta riippumattomat navigointimallit | 12 |
| 3.3 | HRI:n vaatimukset navigointiin | 13 |
| 3.3.1 | Robotin sosiaalinen liike | 13 |
| 3.4 | Opasrobotin navigointimalli | 15 |
| 4 | Kehitysalusta | 15 |
| 4.1 | Yleiskuvaus | 16 |
| 4.2 | Runko | 17 |
| 4.3 | Anturit | 18 |
| 4.3.1 | Laserskanneri | 18 |
| 4.3.2 | Infrapuna-anturit | 19 |
| 4.3.3 | Kamera | 19 |
| 4.3.4 | Puskurikytkin | 20 |

| | | |
|-------|---|----|
| 4.3.5 | Gyroskooppi | 21 |
| 4.4 | Moottorit ja voimansiirto | 21 |
| 4.5 | Kinematiikka | 22 |
| 4.6 | Ohjelmointiympäristö | 24 |
| 4.6.1 | Robotino View -projekti | 25 |
| 4.6.2 | Pääohjelma | 25 |
| 4.6.3 | Aliohjelma | 26 |
| 4.6.4 | Muut ohjelmointitavat | 26 |
| 4.7 | Ohjelmistorakenne | 27 |
| 5 | Kehitysprojekti | 28 |
| 5.1 | Projektin kuvaus | 29 |
| 5.2 | Liikkuminen ja toiminta | 29 |
| 5.3 | Toiminta-aika | 30 |
| 5.3.1 | Automaattinen latauspiste | 30 |
| 5.4 | Paikannus | 31 |
| 5.5 | Käyttäjäraja-alue | 31 |
| 5.6 | Turvallisuus | 32 |
| 5.6.1 | Mekaaniset vaarat | 32 |
| 5.6.2 | Varolaitteet | 32 |
| 5.7 | Toiminta | 33 |
| 6 | Yhteenveto | 33 |
| | Lähteet | 35 |
| | Liitteet | |
| | Liite 1. Opasrobotin prototyypin Robotino View -navigointiohjelmien pääosat | |

Lyhenteet ja termit

| | |
|----------|--|
| HRI | Ihmisen ja robotin välinen vuorovaikutus |
| MMI | Monimediakäyttöliittymä |
| GUI | Graafinen käyttäjärajapinta |
| SLAM | Reaaliaikainen paikoitus ja kartoitus |
| VFF | Vektorien voimakenttämalli |
| VFH | Vektorien voimahistogrammimalli |
| USB | Universaali sarjaväyläarkkitehtuuri |
| PWM | Pulssinleveysmodulaatio |
| IP | Verkon tiedonvälitysprotokolla |
| URL | Tiedon kohteen kertova merkkijono |
| API | Ohjelmointirajapinta. |
| HTML | Hypertekstin merkintäkieli |
| UDP | Varmentamaton tiedonvälitysprotokolla IP -verkossa |
| Firmware | Laiteohjelma, tässä sovelluksessa sulautetun järjestelmän kiinteään muistiin tallennettu laitteen toiminnalle tärkeä ohjelma |
| RVW | Robotin ohjelmointiympäristön projektien tiedostomuoto |

1 Johdanto

Ensimmäisten teollisten hydraulirobottien käyttöönotoista 1960-luvulla lähtien on tiedekentällä tehty ahkerasti työtä robottiteknologian kehittämiseksi edelleen. Aluksi keskityttiin työtehtäviin, jotka olivat toistuvia ja yksinkertaisia - robotti nähtiin tuotantolaitteen osana. Teollisuusrobottien käyttökohteet ovat vähitellen muuttuneet joustavammiksi ja monimutkaisemmiksi. Ne yhdistelevät automaattista käyttöä ihmisten suorittamaan manuaaliajoon, robottiohjaimen sisältäessä tuotannonohjauksen mukaan valittuja työtehtävämoduuleita. Vuosien kuluessa robotiikkakehitys on levinnyt teollisuuden liukuhihnasovelluksista myös muihin käyttökohteisiin, kuten yhteiskunnallisen lähilogistiikan palveluihin, joissa vaaditaan robotilta osin itsenäistä toimintaa. Siirryttäessä teollisuuden toistuvista tehtävistä kohti autonomista robotiikkaa käyttäjärajapintojen ja robottien interaktiivisuuden merkitys korostuu.

On todettu, että robottien kognitiivisuuden lisääntyessä tulevaisuudessa, niiden potentiaali helpottaa tuotannollisia haasteita kotitalouksissa, maataloudessa, koulutuksessa ja sosiaali- ja terveysalalla on suuri. Yhteiskunnallisen robotiikan kehittäjien yhteinen tavoite on tuottaa robotti, joka on osa ihmisen jokapäiväistä elämää. Tämän tavoitteen saavuttamisesta on olemassa hyviä esimerkkejä - monissa kodeissa on jo käytössä robotti-imureita ja robottiruohonleikkureita. [1]

Tiedekeskus Heureka aloitti Festo Oy:n kanssa tammikuussa 2013 yhteistyöhankkeen, jossa kehitetään yhteiskunnallisessa tehtävässä toimivaa robotiikkaa. Haluttiin luoda autonominen, vapaasti liikkuva opasrobotti, joka palvelee julkisessa näyttelytilassa vierailijoita. Kohteen julkistus tapahtuu kesäkuussa 2014.

Toimivan kokonaisuuden luomiseen liittyy monia teknisiä haasteita. Yksi niistä oli robotin paikoitus ja navigointi, johon tässä insinööriyössä keskitytään. Sijainnin määrittelyn, määränpään löytämisen ja törmäilyn estämisen lisäksi robotin liikkumisen ja fyysisen käytöksen tulisi tukea sen tavoitetta olla helposti lähestyttävä ja käytännöllinen tukitoiminto näyttelyssä. Navigointi ja paikoitus osakokonaisuutena on aihe, johon nivoutuu teknisen toteutuskokonaisuuden lisäksi vahvasti robotin sosiaalisen käytöksen määrittelyä.

1.1 Projektin osapuolet

1.1.1 Festo Oy

Festo AG & Co KG on vuonna 1925 perustettu Saksalainen perheyrius, jolla on 59 myyntiyhtiötä eri maissa sekä liiketoimintaa 186 maassa. Festo Oy on Suomen myyntiyhtiö. Festo työllistää noin 14 300 ihmistä maailmanlaajuisesti. Feston ensimmäiset tuotteet olivat puunkäsittelytyökaluja. 1950-luvulla Festo alkoi tuottaa paineilmalaitteita sahateollisuuteen, ja automaatioteknologian kehittyessä tulevina vuosikymmeninä Festo laajeni myös muille teollisen tuotannon aloille ja kehittyi yhdeksi maailman johtavista teollisuuden automaatioteknologian sekä ohjaustekniikan toimittajaksi. [18,19]

Vuonna 1965 perustettiin Festo Didactic GmbH & Co KG, joka on kouluttamiseen, konsultointiin, oppimisympäristöihin ja oppimissisältöihin erikoistunut maailmanlaajuisesti toimiva yhtiö. Suomessa Festo Didactic on toimittanut oppimisympäristöratkaisuja kaikille koulutusasteille. Festo Didactic valmistaa ja toimittaa projektissa käytettävän robottilaitteiston. [20]

1.1.2 Tiedekeskus Heureka

Heureka on Vantaalla 1989 toimintansa aloittanut tiedekeskus. Sen tavoitteena on on tieteellisen tiedon ymmärrettävyyden lisääminen ja tiedeopetuksen menetelmien kehittäminen. Tiedekeskuksen toiminnasta vastaa Tiedekeskussäätiö. Tiedekeskussäätiö on laajapohjainen yhteistyöjärjestö, jonka toimintaan osallistuvat Suomen tiedeyhteisö, opetuselämä, elinkeinoelämä, valtiovalta ja paikallishallinto. Tiedekeskuksen kokonaispinta-ala on 8200 neliometriä, josta näyttelytilaa on 2800 neliometriä. Keskimääräinen kävijämäärä on ollut vuosittain noin 280 000 kävijää, joista yli puolet on perheitä ja noin neljännes koululaisia. Tiedekeskuksen keskeinen sisätila on 14 metriä korkea sylinterimäinen näyttelyhalli. Opasrobotijärjestelmä tullaan alustavasti sijoittamaan tähän noin 1000 neliömetrin tilaan. Heureka vastaa projektissa robotijärjestelmän mediasisällön tuotannosta. [21]

2 Opasrobotin viitekehys

2.1.1 Robotin stereotyyppi

Karel Capek teki robotti-sanan tutuksi vuonna 1921, mutta ihmisen kaltaisia koneita on visioitu jo tuhansia vuosia. Viimeisen vuosisadan aikana antropomorfiset koneet ovat tulleet suurelle yleisölle tutuiksi tieteiskirjoissa ja kaunokirjallisuudessa, sarjakuvissa, radiokuunnelmissa, elokuvissa ja TV-sarjoissa. Fiktiivisten robottien suosio osoittaa, että ihmiset ovat vastaanottavaisia sille idealle, että jonain päivänä robotit ovat osa ihmisen jokapäiväistä elämää. Kehitystyön alkuvaiheessa visiot robotin toiminnoista olivat teknisesti laajalla pohjalla johtuen monipuolisista populäärikulttuurin fiktiivisten robottien tarjoamista vaikutteista. [1]

Huolimatta siitä, että autoteollisuudessa on jo yksi robotti kymmentä työntekijää kohden, on vielä pitkä matka siihen, että oikeat robotit tekevät samoja tehtäviä kuin tieteisfiktio esikuvansa. Ihmisten odotukset robottien kognitiivisten ominaisuuksien toteutusmahdollisuuksista ovat varsin korkealla. Asiaa tutkittiin projektin alkuvaiheessa peruskoulun kolmasluokkalaisten näkemysten keräämisellä. Noin kymmenvuotiaat lapset ovat suuri kohderyhmä näyttelyssä. Lasten osallistuminen suunnitteluun alkuvaiheessa oli kehitysprojektin ja tiedekeskus Heurekan luonteelle sopivaa. Kuvassa 1 on esiteltynä oppilaiden suunnittelemaa ja kokoonpanemaa mielestään toteutuskelpoisia mobiilirobotteja, joiden päätehtävänä olisi oppaana toimiminen. [1,2]

Robotin tekninen toteutus voi olla hankalaa kokonaistoiminnan kannalta tärkeissä toiminnoissa, jotka asiaan perehtymätön ihminen näkee itsestäänselvyyksinä. Tällaisia toimintoja voivat olla esimerkiksi paikoituksella varustetut monen vapausasteen toimilaitteet, navigointi ja paikoitus, äänen tai puheen tunnistaminen sekä tuottaminen ja konenäkö. Teknisten toteutusmahdollisuuksien yliarviointia kehitysprojektin puitteissa tapahtui sekä kolmasluokkalaisten määrittelemissä roboteissa, että pienissä määrin tiedekeskus Heurekan alustavissa robotin toimintojen määrittelyssä. [1]



Kuva 1. Harjavallan Pirkkalan koulun oppilaiden näkemyksiä opasrobotin toteutuksesta

2.1.2 Kehitysympäristö yhteiskunnallisen robotiikan alalla

Robotiikka tuotantoteollisuuden robotiikan ulkopuolella on vasta syntymässä oleva teollisuudenala. Tällä hetkellä se perustuu urauurtavien automaattilajien lähinnä yritysten erikoissovelluksiin toimittamien robottien, innovatiivisten lelujen, sekä koulutukseen ja harrastustoimintaan valmistettujen robottien tuotantoon. Se on myös erittäin hajanainen teollisuudenala - standardeja ja alustoja on yhtä monta kuin on robotinvalmistajia. Projektit ovat monimutkaisia, edistymisen on hidasta, ja todelliset käytännön sovellukset ovat harvassa. Robotiikan alalta julkaistaan jatkuvasti tutkimuksia ja robottien kehitys alkaa vähitellen perustua sille, että valmiita robotteja käytetään robotiikan jatkokehittelyyn, aina uuden erityisesti käyttökohteeseensa rakennetun robotin rakentamisen sijasta. Perustutkimus robotiikassa keskittyy tällä hetkellä parhaiden navigointialgoritmien ja toimintatapojen kehittelyyn. [1,2]

Toteutusten kustannustaso on yleisesti korkea, koska robotti vaatii paljon anturointia, sekä paljon ohjelmointityötä. Kun tietokoneiden laskentateho kasvaa, yhteisten toimintatapojen yleistyessä, sekä laitteistojen ja anturien tullessa tulevaisuudessa edullisemmiksi, ohjelmointityö ja sulautettujen järjestelmien tuntemuksen puute on suurimpia tekijöitä sille, ettei robotiikka yleisty nykyistä nopeammin. [1]

2.2 Aikaisemmat opasrobotisovellukset

Opasrobotisovelluksia on tehty maailmalla useita. Se on yksi ensimmäisistä todellisista sovelluskohteista palvelurobotille. Museot ja näyttelyt ovat roboteille ja niiden kehittäjille hyvin kiehtova ympäristö. Se mahdollistaa hallitun sisätilojen mallinnuksen, autonomisen navigoinnin ja paikoituksen sekä HRI:n kehittämisen ja tutkimisen. Ensimmäiset opasrobotisovellukset otettiin käyttöön 1990-luvun loppupuolella. [11]

Heurekan opasrobotiprojektin onnistumisen kannalta on tärkeää, että sovelletaan laajasti käyttöön otettuja ja koeteltuja toimintamalleja, joista löytyy paljon kehitystyötä tukevaa materiaalia. Tässä kappaleessa on eritelty kolme yliopistojen tai julkisten tahojen kehittämää robottijärjestelmää ja yksi yksityisen yrityksen kehittämä järjestelmä. Aikaisemmin tehtyjen projektien perusteella on havaittavissa, että julkisten hankkeiden resurssit on käytetty sisällön ja toimintojen tuottamiseen, kun taas yksityisten yritysten hankkeiden resurssoinnissa on käytetty paljon myös ulkoisen vakuuttavuuden varmistamiseen.

2.2.1 Rhino ja Minerva -robotit

Vuonna 1997 otettiin Deutsches Museumissa, Saksan Bonnissa käyttöön Rhinoniminen robotti (kuva 2). Se oli ensimmäinen virallinen opasrobotiksi suunniteltu robotisovellus. Robotin suunnitteli ja rakensi työryhmä, johon kuului tutkijoita Bonnin yliopistosta, Aachenin teknisestä yliopistosta ja Carnegie Mellon -yliopistosta. Sen tehtäviin kuului interaktiivisten tutustumiskierrosten tarjoaminen museossa kävijöille, ja virtuaalisen läsnäolon mahdollistaminen robotin web-käyttöliittymän kautta. Sovelluksen kohde aiheutti monenlaisia haasteita, joita ei oltu aikaisemmissa tutkielmissa kohdattu. Aikaisemmat tutkimukset olivat enimmäkseen sijoittuneet hyvin kontrolloituihin ja toimistomaisiin tiloihin. Kaksi ensisijaista tavoitetta olivat turvallinen navigointi ihmisiä täynnä olevassa ympäristössä, sekä intuitiivinen ja houkutteleva vuorovaikutus ihmisten kanssa. Turvallinen navigointi oli Rhino -projektille tärkeää jatkokehityksen kannalta. Robotti navigoi keskimääräistä kävelyvauhtia törmäilemättä ihmisiin tai esineisiin. [11,12]



Kuva 2. Rhino -robotti

Seuraavat seikat olivat haasteita robotin käytölle:

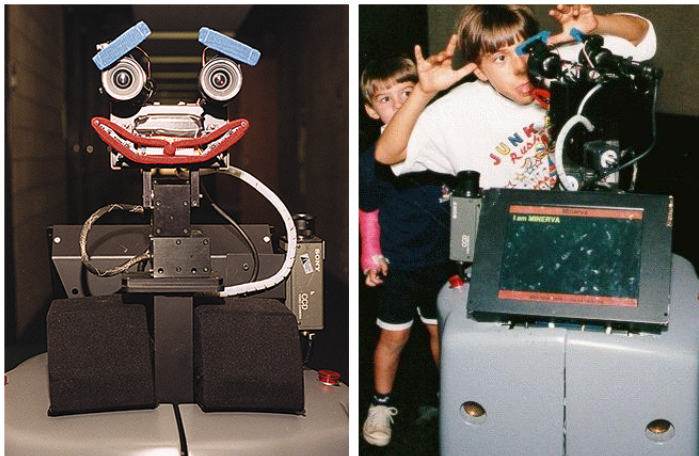
- Robotin piti usein navigoida erittäin tiheän väkijoukon läpi
- Usein ihmiset, tahallaan tai vahingossa, peittivät robotin anturoinnin pitkiksikin ajoiksi
- Näyttelyssä oli esineitä, joita ei voinut havaita kaikilla antureilla (lasi, teräsvaijereilla katosta riippuvat rakenteet)
- Museon kalustus vaihtoi usein paikkaa, pienikin muutos vaikutti kartoitukseen (tuolit, pöydät)
- Kävijät halusivat tahallaan aiheuttaa robotille kolareita.

Minerva-robotti oli jatkokehityshanke Rhino -robotille, kehittäjinään tutkijaryhmä Bonn ja Carnegie Mellon -yliopistoista. Se toimi kaksi viikkoa Smithsonian's National Museum of History -museossa Washingtonissa Yhdysvalloissa keksintöjä ja innovaatioita esittelevässä näyttelyssä kesällä 1998. Toimintansa aikana sillä oli tuhansia asiakkaita. [11,13]

Minerva-projekti käsitteli hyvin samankaltaisia asioita kuin aikaisempi Rhino -projekti:

- Navigointi ja paikoitus dynaamisissa ympäristöissä
- Lyhyen aikavälin vuorovaikutus ihmisen kanssa
- Virtuaalinen etäkäyttö.

Minerva-projektissa tehostettiin robotin navigointia lisäämällä siihen inhimillinen kommunikointitapa (kuva 3). Minerva-robotin suuta ja kulmakarvoja edustavat toimilaitteet, sekä kameroiden suuntaus viestittivät edessä olevalle ihmiselle esimerkiksi sen, että hänen pitäisi mennä pois tieltä. Se havaittiin tehokkaammaksi tavaksi kuin Rhino-robotissa ollut auton äänimerkki. Myös käyttöliittymää, kartoitusta, paikoitusta ja navigointia muilta osin tehostettiin. Paikoituksessa käytettiin robotista ylöspäin suunnatun kameran robotin liikkuesssa ottamia kuvia, joita verrattiin koko katosta otettuun kuvamalliin. [11,13]



Kuva 3. Minerva-robotti

2.2.2 Urbano -robotti

Urbano -robotin kehittivät Madridin teknisessä yliopistossa kuusihenkinen työryhmä. Urbano -kehitysprojektin päätutkimuslinja oli kehittää autonomisia navigointialgoritmeja mobiiliroboteille, erityisesti samanaikaisen paikoituksen ja kartoituksen (SLAM) ongelmiin. Urbano -robotti on esitellyt useita kohteita pääasiassa tekniikan messutapahtumissa Espanjassa vuosina 2003 - 2007. Urbano toimii siten, että se kartoittaa automaattisesti toiminta-alueensa, jonka jälkeen se voi autonomisesti kulkea mihin tahansa aikaisemmin kartoittamaansa paikkaan. [6]

Urbano-robotti on julkisin varoin rakennettu tutkimus- ja kehitysalusta, jonka jatkokehitys keskittyy 3D-kartoituksen luomiseen, järjestelmän kestävyuden parantamiseen ja rajapintojen yhdenmukaistamiseen.[6]

2.2.3 Toyota TPR Robina -robotti

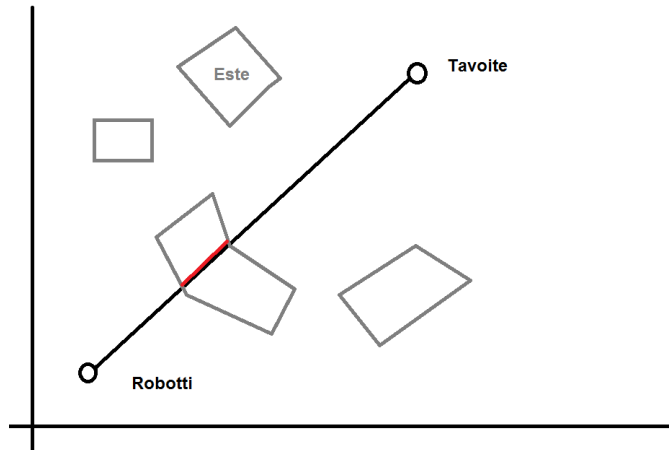
TPR Robina julkistettiin elokuussa 2007. Se kehitettiin Toyotan sairaala-apulaisrobotin ja pyörillä kulkevan DJ-robotin pohjalta. Robina-robotti on suunniteltu toimimaan vastaanottovirkailijana tai opasrobotina. Robotti työskentelee oppaana Toyota Kaikan Exhibition Hallissa Toyotan kaupungissa Japanissa. Robina on noin 1,2 metriä korkea ja painaa noin 60 kilogrammaa. Akunkesto on noin 1 tunti. Ultraäänianturit ja laseretäisyysmittari rungossa antavat tietoa kartoitusohjelmaan, jolla voidaan kartoittaa noin 200 neliömetrin alue. Robotilla on kolmisormiset kädet, joilla se voi esimerkiksi kirjoittaa nimikirjoituksensa. Puheentunnistus ja puhesyntetisaattori mahdollistavat yksinkertaisen keskustelun robotin ja ihmisen välillä. Ongelmia on tuottanut lapsen puheen tunnistus, sekä puheen tunnistaminen meluisassa ympäristössä. [17]



Kuva 4. Toyota TPR Robina -robotti

3 Navigointi ja paikoitus

Suorassa tasossa toimivan mobiilirobotin toiminta-alueen voi esittää kaksiulotteisena koordinaatistona (kuva 5). Robotilla on paikkakoordinaattinsa, ja tavoitteella on paikkakoordinaattinsa. Reitti on näiden kahden koordinaatin välille muodostuva jana. Reitin varrella sijaitsevat esteet voi esittää tilavuuksina koordinaatistossa. Mikäli reitti kulkee jonkin esteen läpi, voidaan sanoa esteen olevan reitillä. Kun robotti lähestyy tavoitettaan, sen pitäisi turvallisen toiminnan takaamiseksi vältellä törmäyksiä paikallaan pysyvien esteiden sekä ihmisten ja muiden dynaamisten esteiden kanssa reittinsä varrella. [7]



Kuva 5. Robotin toiminta suorassa tasossa

Robotiikassa navigoinnin probleeman voisi kiteyttää kolmeen kysymykseen: [2,3]

- Missä minä olen?
- Mihin olen menossa?
- Kuinka minä pääsen sinne?

3.1 Paikoitus

Teoriassa helpoin tapa määrittää robotin sijainti olisi jollakin tavalla mitata sen sijainti suhteessa ympäristöönsä. Tämän toteuttaminen elävässä elämässä ei ole aina mahdollista. Suoraviivainen tapa mitata robotin sijaintia on integroida sen kulkemaa nopeutta ja suuntaa. Tämä tieto voidaan määrittellä, jos tiedetään robotille sen eri vapausasteisiin annetut nopeusohjeet ajan funktiona. Tätä tietoa käytetään yleisesti robotiikassa paikan määrittämiseen. Valitettavasti mobiilirobotiikassa tällä tavalla määritelty sijainti on usein epätarkka johtuen renkaiden tai robottia liikuttavien toimilaitteiden aiheuttamasta liukumisesta. [2,3,5]

Lisäämällä rinnakkaisia mittauksia robotin asemasta voidaan tarkemmin arvioida robotin todellinen sijainti. Usein robotin toiminnassa käytetään myös kalibroivia mittaustapoja, joissa hetkellisesti tiedetään suurella todennäköisyydellä robotin tarkka absoluuttinen sijainti. Tällöin voidaan nollata suhteellisen sijainnimäärittelyn aiheuttama mittavirhe.

3.2 Navigointimallit

Kun tunnetaan robotin asema riittävän tarkasti navigoinnin aikana, voidaan määritellä navigoinnille kaksi vaihtoehtoista päätoimintastrategiaa, suunnitelmallinen navigointi ja reaktiivinen navigointi.

3.2.1 Suunnitelmallinen navigointi

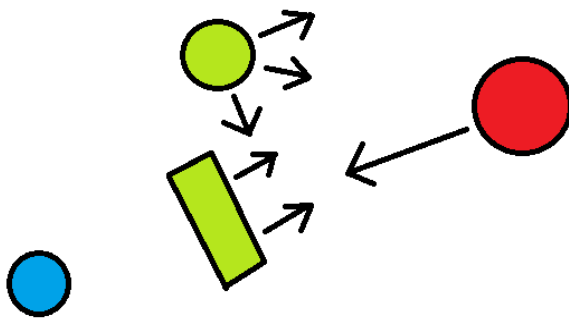
Puhtaasti suunnitelmallisessa navigoinnissa ennalta kartoitetulla alueella toimiva robotti määrittelee reittinsä ennakkoon, ja seuraa reittiä kohteeseensa. Etenemissuunnassa liikkuvien kohteiden kanssa törmäily on estettävä reaktiivisesti, esimerkiksi pysähtymällä jos jokin asia tukkii reitin. Toimintatapa on varma käyttökohteissa, joissa dynaamisia esteitä tai usein muuttuvia järjestelyitä ei ole. Järjestelmä on jäykkä, ja toiminta pysähtyy kokonaan, mikäli ennakkoon kartoitettuja paikallaan pysyviä esteitä siirrellään. Tätä ominaisuutta voidaan pyrkiä ehkäisemään lisäämällä suunnitelmalliseen navigointiin reaktiivisen navigoinnin toimintamalleja. Kartoituksella ja sen päivityksellä on suuri merkitys suunnitelmallisen navigoinnin onnistumisessa. [4,5,7]

3.2.2 Reaktiiviset navigointimallit

Erilaisia reaktiivisia navigointimalleja on monia. Ne perustuvat puhtaimmillaan sille periaatteelle, ettei robotti tiedä kohteen ja lähtöpisteen välisistä esteistä mitään matkaan lähtiessään. [4]

Ensimmäisiä lähestymistapoja 1980-luvun lopulla reaktiivisen väistelyn toteuttamiseen oli kulmantunnistusmenetelmä. Tällä algoritmilla etsittään ensin kulma eteen sattuneesta kohteesta. Kohteen kulma tunnistetaan robotin kolmesta etäisyysanturista keskimmaisella. Tunnistuksen jälkeen robotti tekee edestakaista kääntymisliikettä tunnistaakseen kumman kulmasta lähtevän sivun kulma on suurempi suhteessa robotin vastaiseen tasoon. Robotti liikkuu sinne, missä kulma on suurempi eli sinne, missä on enemmän tilaa liikkua. Tämä menetelmä on hidas ja riippuvainen siitä, että kohteesta löytyy etäisyysanturin lähettämää signaalia riittävästi hajauttava kulma. [4,5,7]

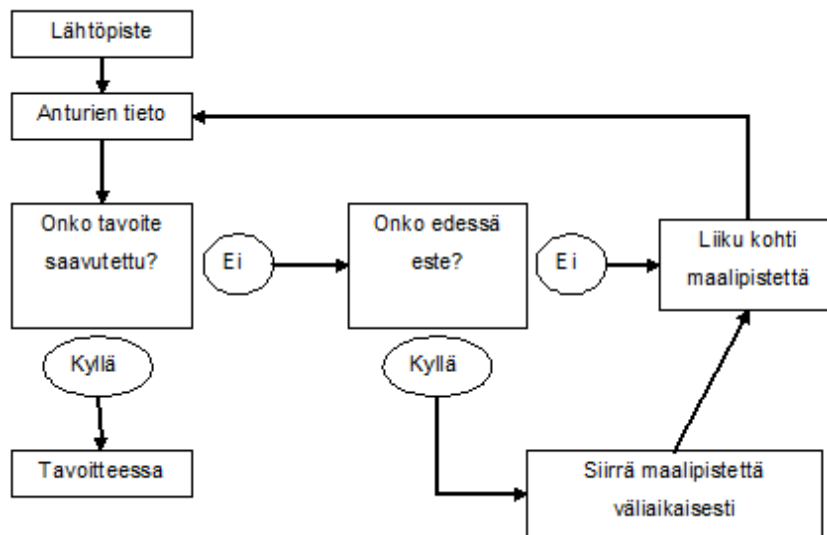
Potentiaalikenttämallit perustuvat siihen, että kun kohde asettaa robotille suunnassaan puoleensavetävän voimavektorin, niin etäisyysantureilla etenemissuunnassa mitatut esteet antavat tasostaan pystysuoran hylkivän voimavektorin (kuva 6). Laskemalla näiden vektorien yhteisvaikutus saadaan robotille etenemissuunta. Etäisyys esteeseen saadaan esimerkiksi etäisyysanturilta. Esteiden liikuessa repulsiivinen kokonaisvoima muuttuu siten, että robotti ohjaa esteistä poispäin jatkuvasti. Menetelmä soveltuu dynaamiseen ja reaaliaikaiseen tilanteeseen. Menetelmä ei ota kantaa esteiden liikkumiseen vaan yksinkertaisesti määrää robottia hylkimään esteitä silloin kun ne ovat havaintosäteellä eli se ei luo dynaamisia malleja. Algoritmi ei yksinään toimi täydellisesti, vaan se voi suurella todennäköisyydellä aiheuttaa robotille tilanteen, jossa se jää jaksolliseen edestakaiseen liikkeeseen pääsemättä lähemmäs kohdettaan, esimerkiksi pyrkiessään ohi tietyn muotoisesta esteestä. Tällainen tilanne voi tapahtua kulmissa ja umpikujissa. [4,5,7]



Kuva 6. Potentiaalikenttämalli. Kohde on merkitty sinisellä.

Varhaisia potentiaalikenttiin perustuvia väistelymalleja ovat VFF ja VFH, vektorien voimakenttä- ja voimahistogrammimallit. VFF perustuu histogrammin keräämiseen etäisyysmittaustiedoista. Robotin tielle sattuvat esteet, jotka muodostavat suuria alueita robotin lähellä ja myös ajallisesti pysyvät siellä ja tuottavat voimakkaita hylkiviä voimia. VFF-malleille pohjautuen on luotu umpikujia vältteleviä VFH-malleja, jotka tarkkailevat kerättyä hirstogrammia ja ennakoita luodun mallin perusteella osaavat valita oikean reitin. Näihin malleihin on usein myös sisäänrakennettu dynaamisia ja tietyn robotin kinematiikkamallin huomioonottavia algoritmeja. Nämä algoritmit perustuvat sille, että ainakin robotin etenemissuunnassa voidaan mitata etäisyyksiä robotin ja esteiden välillä. [2,5]

On myös algoritmeja, jotka perustuvat jonkin ennalta määritellyn esteen kiertämiseen tähtäävän toiminnan tekemiseen silloin kun este havaitaan. Yksi ratkaisu on virtuaalisesti asettaa robotin maalipiste väliaikaisesti havaitun esteen tangentin suunnassa sivuun. Myös reitillä pysymisen, tai tavoitteen lähestymiskulman määrittelyssä tavoitepisteen virtuaalinen siirtäminen on mahdollista. [7]



Kuva 7. Maalipisteen siirtoalgoritmin lohkokaavio

Yleisesti reaktiiviset navigointimallit sopivat parhaiten rakenteellisesti kevyeen ympäristöön. Ottaessaan huomioon ympäristön ja robotin dynamiikan, ne sopivat erinomaisesti esimerkiksi ihmisten seassa liikkumiseen tiloissa, jossa on paljon väistämistilaa. Mikäli liikkumatilassa on paljon rakenteita ja kulmia, pelkällä reaktiivisella navigoinnilla ei tule toimeen. Tähän tarvitaan lisäksi ennalta kartoitettua ja suunnitelmallista liikkumismallia. [2,5]

3.2.3 Robotin asemasta riippumattomat navigointimallit

Jos ei tarkalleen pystytä määrittelemään robotin asemaa liikkeen aikana, vaaditaan navigoinnilta myös paikoituksen varmistamiseen tai paikoituksen tarpeen poistamiseen tähtääviä toimintoja. Tällainen toiminto voi olla esimerkiksi tunnettujen fyysisten rakenteiden seuraaminen tavoitteen löytämiseksi. Tätä toimintatapaa käytetään muun muassa teollisuuden vihivaunusovelluksissa, joissa vaunu voi seurata lattiaan merkittyjä viivoja, tai mekaanisia raiteita. Se on kustannustehokas ratkaisu, joka vaatii

staattisen ja hallitun ympäristön. Vapaasti liikkuvissa sovelluksissa tunnetun fyysisen rakenteen seuraaminen voidaan lisätä osaksi robotin poikkeuskäyttäytymistä kuten tapauksissa, joissa robotin paikoitus ei toimi. [5]

3.3 HRI:n vaatimukset navigointiin

Autonomisten palvelurobottien kehityksessä yksi suurista haasteista on aikaansaada toimiva ihmisen ja robotin välinen vuorovaikutus, HRI. Tutkimuskohteena se on solmukohta, joka haarautuu kaikkiin yhteiskunnallisen robotiikan käyttökohteisiin, ja se on otettava huomioon kaikissa robotin toiminnoissa. HRI on kokonaisuus, johon liittyy psykologiaa ja sosiaalitieteitä, toisaalta mekaniikkaa, automaatiota ja tietojenhallintatiedettä. Sen tavoitteena on tuottaa turvallinen, miellyttävä ja tehokas rajapinta robotin tekemän työn ja samassa tilassa toimivan ihmisen välille. [2,3]

On koulukuntaeroja siinä, mikä on optimaalinen HRI. Samaa tavoitetta lähestytään kahdesta suunnasta, joko sopeuttamalla robotteja ihmisten maailmaan tai sopeuttamalla ihmisiä robottien maailmaan. Joidenkin tutkimusten mukaan paras HRI on sellainen, jossa robotti käyttäytyy mahdollisimman inhimillisesti, ja myös muistuttaa ulkonäöltään ihmistä. Toisaalta on tutkimusta siitä, että ihminen käyttäytyy sosiaalisesti kaikkea kohtaan, myös esineitä, ja näinollen omaksuu erilaisia toimintamalleja. Tällöin HRI-kehityksen tavoitteena on rajapinta, jonka ihminen omaksuu mahdollisimman helposti, eli käyttöliittymä on mahdollisimman intuitiivinen. [2,3]

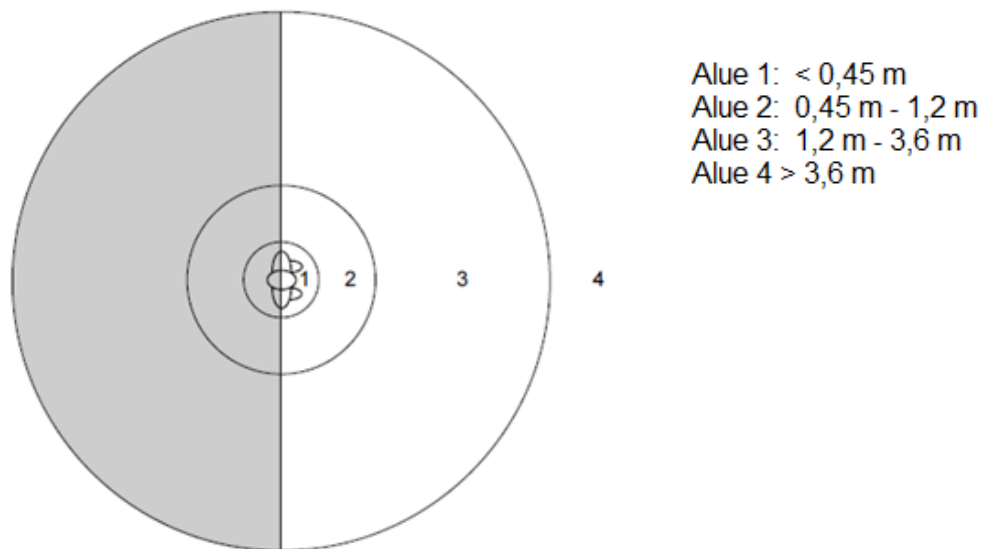
Heureka opasrobotin HRI:n tavoitteeksi päätettiin ottaa mahdollisimman intuitiivinen ja helppo kosketusnäytöllä toteutettu GUI yhdistettynä inhimilliseen navigointiin. Robotin ulkonäköön tultaisiin toisaalta ottamaan osallistava ja inhimillistä viestintää korostava kehityssuunta.

3.3.1 Robotin sosiaalinen liike

On tärkeää, että ihmiset hyväksyvät robotin ympäristössään. Jos ihminen pelkää tai varoo liikkuvaa robottia, se ei voi suorittaa tehtäväänsä tehokkaasti. Alueellinen käyttäytyminen on tärkeä osa opasrobotin navigointia. Tiedeyhteisöissä tehtyjen tutkimusten perusteella robotin liikkeen imitoidessa ihmisen liikettä, hyväksytään se helpommin. Yksi inhimillisen liikekommunikaation osa-alue on yksilön tilan tarve. [2,3]

Kuva 8 mallintaa ihmisen tilantarvetta. Ihmisen sosiaalinen tila koostuu viidestä vyöhykkeestä: [2]

- Harmaa lohko on ihmisen epämukavuusaluetta. On suositeltavaa, ettei robotti liiku tällä alueella ilmoittamatta itsestään
- Alue 4 on julkista tilaa. Robotti voi liikkua vapaasti
- Alue 3 luokitellaan ihmisen sosiaaliseksi alueeksi. Robotin tulee rajoittaa nopeuttaan ja pysytellä ihmisen havaittavissa. Robotti ei saa tällä alueella lähestyä ihmistä kohtisuoraan
- Alue 2 on ihmisen henkilökohtaista aluetta. Robotin on sallittua toimia tällä alueella vain kommunikoidakseen ihmisen kanssa
- Alue 1 on ihmisen intiimialuetta. Robotti ei saa tulla tälle alueelle, ja sen on mahdollisuuksien mukaan poistuttava tältä alueelta.



Kuva 8. Ihmisen sosiaaliset vyöhykkeet

Lisäksi robotin tulee täyttää seuraavat vaatimukset:

- Robotin liikkeen tulisi olla sulavaa ja ennalta arvattavaa
- Robotti ei saa olla törmäävällä liikeradalla ihmisen kanssa
- Robotin koko tulee olla pienempi kuin ihminen
- Robotin tulee pystyä näyttämään oma tilansa.

Kun todetaan, ettei opasrobotti voi tarkasti määritellä ihmisen rintamasuuntaa reittinsä varrella tai riittävän tarkasti tunnistaa ihmistä yleensäkin, voidaan johtopäätöksenä määritellä robotille sopivat varoetäisyydet, joita se käyttää navigoidessaan näyttelyssä. Robotin tulee pysähtyä tai kiertää esteet vähintään 1,2 metrin etäisyydellä ja rajoittaa nopeuttaan rauhallisemmaksi, kun se havaitsee esteen 3,6 metrin etäisyydellä etupuoolellaan. Robotin reitille tulevia esteitä ei saa lähestyä kohtisuoraan alle 3,6 metrin etäisyydellä. Voidaan myös määritellä varmoja pisteitä, joissa tiedetään olevan eloton este joka vaikuttaa tehtyihin mittauksiin. [2]

3.4 Opasrobotin navigointimalli

Väistelymallin tulee olla Heurekan opasrobottisovelluksessa varovainen. Aggressiivinen väistelyliike voi aiheuttaa reitiltä eksymisen, koska väistelystä johtuva pitkittyvä kuljettu matka lisää kumulatiivisesti paikoituksen virhettä. Tämä vaarantaa perille pääsemisen ja edelleen robotin toiminnan. Lisäksi robotille vaadittu sosiaalinen vyöhykekäyttäytyminen tulee turvata. Aikaisempien opasrobottien navigointia verrattaessa on havaittu, että nopeampi ja teknisempi navigointialgoritmi ei välttämättä ole toiminut niin hyvin sosiaalisessa toimintaympäristössä kuin rauhallinen ja sosiaaliseen kanssakäymiseen keskittyvä toimintamalli. Navigointia käsittelevissä tutkimuksissa on myös usein todettu, että sopivin väistelyalgoritmi on hyvin tapauskohtainen. [5,7,11]

4 Kehitysalusta

Robottijärjestelmäkokonaisuuden rakentaminen alusta asti ei olisi ollut projektin resurssien puitteissa mahdollista, joten järjestelmän alustaksi valittiin kehitystyön alkuvaiheessa Festo Didactic Robotino[®]. Se on avoimen lähdekoodin mobiilirobottialusta, joka on suunniteltu sulautettujen järjestelmien tutkimus- ja koulutuskäyttöön. Robotinosta on kolme erilaista versiota. Versiot 1 ja 2 ovat ulkoisesti samanlaiset (kuva 9). Eroavaisuuksia löytyy järjestelmän rakenteesta ja sulautetun järjestelmän ominaisuuksissa. Robotinon kolmas sukupolvi julkaistaan asteittain vuoden 2013 lopulla ja 2014 alussa. Heurekan opasrobottijärjestelmän toteutuksessa tullaan käyttämään Robotino 3:a. Opasrobotin prototyyppi, joka rakennettiin keväällä ja syksyllä 2013, on modifioitu Robotino 2.



Kuva 9. Robotino 1 ja 2 ulkoasu, Robotino 3 varustettuna keskitolpalla

4.1 Yleiskuvaus

Robotino on kokonaisuutena toimitettava, suuntaamattomalla, holonomisella alustan geometrialla rakennettu mobiilirobotti. Kolme ajomoottoria mahdollistaa liikkeen kaikkiin suuntiin, eteen, taakse ja sivuille. Robotti voi myös kääntää rintamasuuntaansa paikallaan tai liikkeessä riippumatta muusta liikkeestään. Robotti on vakiona varusteltu web-kameralla ja useilla erilaisilla antureilla: analogisilla etäisyysantureilla, törmäyspuskurianturilla ja sähkömoottoreiden inkrementtiantureilla, joissa on valmis rajapinta robotin sulautettuun käyttöjärjestelmään. Anturien ja laitteiden lisäys on tehty helpoksi USB:n tai I/O liittimen välityksellä. [9,10]

Robotinon ohjauslaitteisto koostuu robotin ohjauskorttiin sulautetusta PC:stä ja IO-kortista. Normaalisti Robotinoa ohjataan ja ohjelmoidaan liittymällä sen lähiverkkoon ja ajamalla luotuja ohjelmia lähiverkossa. Ohjelma voidaan myös ladata Robotinoon ja ajaa suoraan Robotinon käyttöjärjestelmässä, jolloin robottiohjelmaa ajavaa yhteyttä ulkoiseen ohjaimeen ei tarvita. [8,9,10]

4.2 Runko

Robotin runko on laserhitsattua ruostumatonta terästä. Runkoon asennetaan seuraavat toiminnalle tärkeät komponentit:

- Akusto
- Moottorit, voimansiirto ja renkaat
- Infrapuna-anturit
- Törmäyspuskurianturi
- Mahdolliset latauslaitteistot
- Komentosilta
- Pystyolppa korkealle asennettavia instrumentteja varten.

Opasrobotin prototyyppi on rungoltaan modifioitu Robotino 2. Sen runkoon on kiinnitetty pystysuunnassa noin 120 senttimetriä pitkä alumiiniprofiili, johon on kiinnitetty korkeammalle asennettavat mittalaitteet ja käyttöliittymälaite (kuva 9). Robotino 3:ssa on rungon keskelle kiinnitettävä ruostumattomasta teräksestä valmistettu pystyolppa, johon voidaan kiinnittää tarvittavat instrumentit. [9,10]



Kuva 10. Opasrobotin prototyyppi

Lopullisen opasrobottijärjestelmän runkoon tullaan liittämään muovista tai muusta komposiitista valmistettu ulkokuori.

4.3 Anturit

Robottijärjestelmään sisältyy paljon anturointia ja robotin tiedonkeruuta käyttöympäristöstä. Se on järjestelmän turvallisuuden ja toiminnan kannalta tarpeellista. Robotin toiminta perustuu vapaasti ympäristössä liikkumiseen, joten kaikki aktiivinen ohjaus tulee suorittaa antureiden antaman mittaustiedon perusteella.

4.3.1 Laserskanneri

Robotin opasrobottisovelluksen etäisyyksien mittaukseen käytettävä laserskanneri on Hokuyo URG-04LX-UG01. Se liitetään Robotinon USB-väylään. Skannerissa on valmis rajapinta Robotinon järjestelmään. Käyttökokeissa todettiin, että skanneri on tehokas noin neljän metrin mittausetäisyyksiin asti. [9,10]

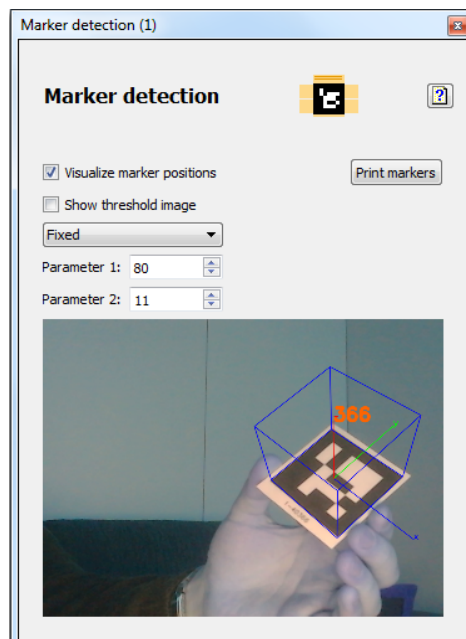
Laseretäisyysmittaus on etäisyydenmittaustapa, joka perustuu laservalopulssin lähettämisen ja heijastumisen välisen ajan mittauksesta laskettuun etäisyyteen. Sama toimintaperiaate pätee suureen osaan optisista etäisyysantureista. Kaksiulotteinen laserskanneri on mittalaite, jossa laseretäisyysmittari pyörii akselinsa ympäri, antaen vaakatason sektorin etäisyydet vektorina aikarasterin jakamana. Pyörimisnopeus anturilla on kuitenkin niin suuri, että yhden pyörähdyksen antamat etäisyysmittaukset voidaan robotin toiminnassa tulkita yhden reaalisen ajanhetken mittaukseksi. Laserskannereita käytetään yleisesti mobiiliroboteissa. Laserskannerin tuottamaa mittatietoa voidaan käyttää kartoitukseen, kohteen havaitsemiseen ja rajaamiseen sekä navigointiin. Laservalo on koherenttia ja kollimoitunutta näkyvää valoa, joten sillä on suhteellisen pitkä kantomatka pienellä lähetysteholla, ja sillä voidaan mitata pinta-alaltaan pieniä kohteita sekä tarkkoja reunoja, koska valon diffraktio on pientä. Lasermittaus on herkkä virheille kohteissa joissa on rakenteita, jotka päästävät lävitseen näkyvää valoa, kuten lasia. [16,2]

4.3.2 Infrapuna-anturit

Infrapuna-antureita käytetään sovelluksessa robotin lähialueen varmistamiseen. Robotinossa käytetyt Sharp GP2Y0A41SK0F -infrapuna-anturit ovat analogisia etäisyysantureita, joiden suurin tehokas mittausetäisyys on noin 4 – 30 senttimetriä. yhdeksän anturia on sijoitettu 40 asteen välimatkoin pitkin robotin alareunan kehää. Ne pysäyttävät robotin etenemisen, jos jokin niistä reagoi robotin etenemissuunnassa. Anturit on kytketty kiinteästi Robotinon IO-korttiin. Infrapuna-antureihin saattavat aiheuttaa virhettä auringonvalo sekä muut infrapunavaloa lähettävät laitteet, kuten kaukosäätimet. Käyttökokeissa havaittiin, ettei antureita voida niiden epävarmuustekijöiden vuoksi käyttää tarkkaan etäisyyden mittaukseen opasrobotin käyttöympäristössä. [9]

4.3.3 Kamera

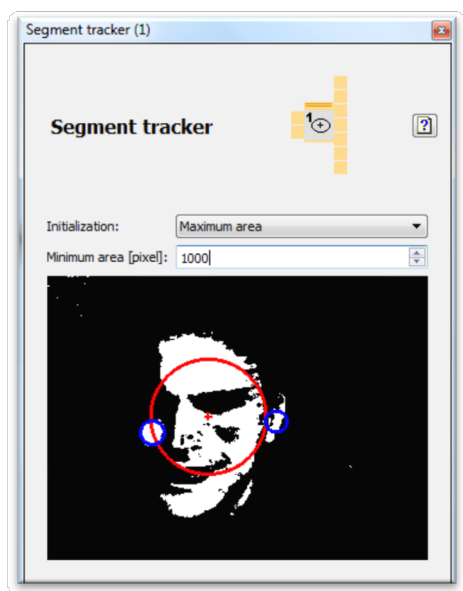
Kameraa käytetään tässä sovelluksessa paikkareferenssipisteiden visualisointiin käytettyjen matriisikoodien lukemiseen (kuva 11). Kamera on kytketty Robotinon USB-väylään. Kameralla on valmis ajuri Robotinon järjestelmään. Robotin navigoinnissa käytetään maalipisteiden opetusmetodia, osin paikoituksen epävarmuustekijöiden vuoksi.



Kuva 11. Matriisikoodin indeksin ja asennon tunnistus Robotinon ohjelmointiympäristössä

Maalipisteet ja niiden väliset reittipisteet listataan robotin maali- ja reittitietokantaan. Maalipisteisiin asennetaan robotin kameran havaintoalueelle matriisikoodi, jonka asento robotin suhteen määritetään konenäöllä, kun robotti saavuttaa maalin lähialueen. Tällä toiminnolla nollataan robotin anturien navigoinnin aiheuttama paikoitusvirhe. [8,9,10]

Kameraa kokeiltiin myös ihonvärin tunnistamiseen ja näin ihmisten tunnistamiseen, mutta käyttökohteen muuttuvat valaistusominaisuudet aurinkoisesta päivänvalosta keinovalon aallonpituuksille muuttivat robotin havaitsemaa ihoaluetta. Myös ihonväriltään erilaiset ihmiset jäisivät tietylle värikaistalle viritetyltä robotilta tunnistamatta. Tämä ei olisi sosiaalisesti hyväksyttävää. Kuvassa 12 robotin konenäköanturi tunnistaa kasvot.



Kuva 12. Tunnistetut kasvot Robotinon ohjelmointiympäristössä

4.3.4 Puskurikytkin

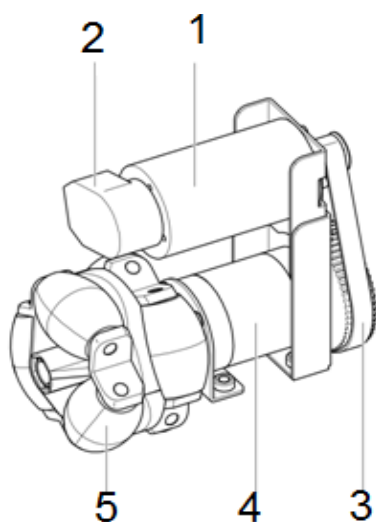
Turvallisuuden takaamiseksi Robotino on varustettu myös sähkömekaanisella puskurikytkimellä. Se kiertää Robotinon alakehää. Mekaaninen kontakti puskurikytkimeen aiheuttaa kahden pitkittäisen, toistensa rinnalla kulkevan metalliliuskan kosketuksen anturin sisällä. Kontaktitieto luetaan robotin IO-kytkennän kautta. [9,10]

4.3.5 Gyroskooppi

Koska robotin sähkömoottorien inkrementtianturit ovat epätarkkoja mittalaitteita robotin paikan määrittämisessä, tarvitaan lisäksi gyroskooppi. Gyroskooppi mittaa kulmanopeuksia robotin ohjailun aiheuttaman impulssimomentin muutoksen perusteella. Cruzcore XG1000 on elektroninen ja digitaalinen gyroskooppi joka voidaan liittää suoraan Robotinon USB-väylään. Erillinen gyroskooppi tarvittiin vain opasrobottijärjestelmän prototyyppiin - Robotinon 3:nnessä versiossa gyroskooppi ja lisäksi kiihtyvyyssanturointi on sisäänrakennettu robotin ohjauskorttiin ja ohjelmointirajapintaan sähkömoottorien inkrementtianturien kanssa.[9,10]

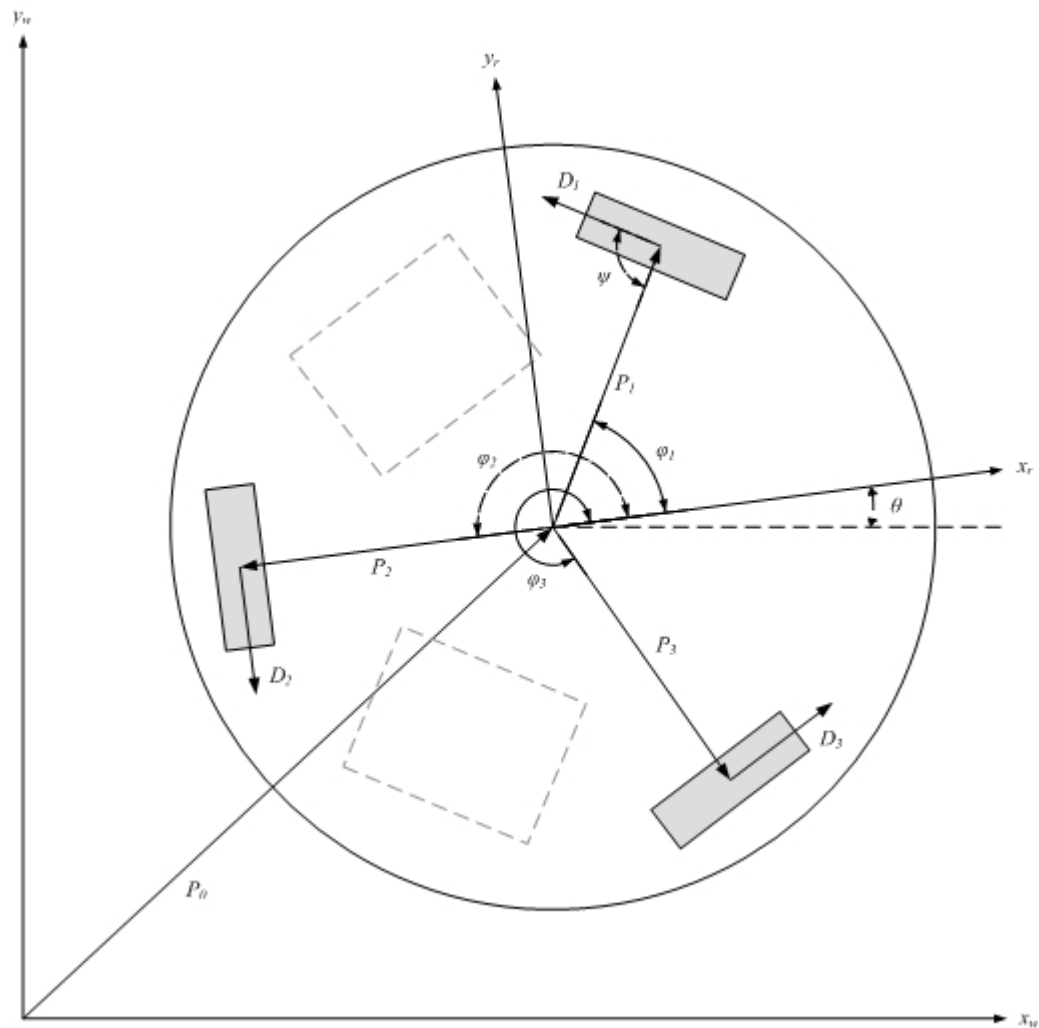
4.4 Moottorit ja voimansiirto

Robotinoa ohjaa kolme toisistaan riippumatonta 24 tasajännite harjamoottorikäyttöyksikköä. Ne on asennettu 120 asteen kulmaan toisiinsa nähden. Planeettavaihteisto ja moottori on kiinnitetty erilliseen robotin runkoon kiinnitettävään laippaan. Moottorin inkrementtianturi mittaa moottorin asennon muutosta ja pyörimisnopeutta tuottamalla moottorin akselin pyöriessä pulssia, jota takaisinkytkettynä PWM -moottorinohjaukseen voidaan käyttää pyörimisnopeuden säätämiseen.[9,10]



Kuva 13. 1: Sähkömoottori, 2: Inkrementtianturi, 3: Hammashihna, 4: planeettavaihteisto, 5: rullapyörä

4.5 Kinematiikka



Kuva 14. Robotin kinematiikan kuvaaja

Robotin on voitava seurata omaa sijaintiaan ympäristöönsä nähden, ja robotin moottorien ohjauksen vaikutus robotin liikkeeseen on tunnettava. Tämä voidaan tehdä määrittämällä robotin pyörien pyörimisen vaikutus robotin liikkeeseen ympäristön koordinaatistossa. [2,3,22]

Kuvassa 14 ovat kuvattuna:

- X_w ja Y_w Koordinaatisto ympäristön suhteen
- X_r ja Y_r Koordinaatisto robotin suhteen
- P_0 Robotin origon asema ympäristön origon suhteen

- $P_{1,2,3}$ Renkaiden keskipisteen suhde robotin keskipisteeseen
- $D_{1,2,3}$ Renkaiden positiivisen pyörimissuunnan yksikkövektorit
- Θ Robotin kiertokulma ympäristön suhteen
- $\Phi_{1,2,3}$ Pyörän akselien kiertokulmat robotin suhteen
- Ψ Pyörän akselin kulma pyörään nähden

Laskennassa oletetaan yksinkertaistussyistä, että pyörien ja maan välinen kitka on ääretön, ja että robotin massakeskipiste on sen geometrisessa keskipisteessä. Näillä oletuksilla voidaan mallintaa pyörien liikkeestä robotin liike ympäristön suhteen. [2,3,22]

Määritellään vektori jossa $v_{1,2,3}$ ovat pyörien pyörimisnopeudet

$$v_w = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}^T$$

Robotin nopeus robotin koordinaatiston suhteen määritellään

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} -\sin(\varphi_1) & \cos(\varphi_1) & R \\ -\sin(\varphi_2) & \cos(\varphi_2) & R \\ -\sin(\varphi_3) & \cos(\varphi_3) & R \end{bmatrix} \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \Omega \end{bmatrix}$$

- Ω robotin pyörimisnopeus
- x_r nopeus x-suunnassa
- y_r nopeus y-suunnassa
- $\omega_{1,2,3}$ pyörien kulmanopeudet
- r pyörän säde
- R pyörän keskipisteen etäisyys robotin keskipisteeseen

Jos halutaan tietää nopeus ympäristön koordinaatiston suhteen, lisätään mukaan Θ

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} -\sin(\theta + \varphi_1) & \cos(\theta + \varphi_1) & R \\ -\sin(\theta + \varphi_2) & \cos(\theta + \varphi_2) & R \\ -\sin(\theta + \varphi_3) & \cos(\theta + \varphi_3) & R \end{bmatrix} \begin{bmatrix} \dot{x}_w \\ \dot{y}_w \\ \Omega \end{bmatrix}$$

4.6 Ohjelmointiympäristö

Robotin navigoinnin ja paikoituksen käsittelevät ohjelmat tehdään Robotino View -ohjelmistolla. Se on ohjelmointityökalu, jolla voi luoda ja suorittaa Robotinon ohjelmaprojekteja. Ohjelmointikieli on natiivi, erityisesti Robotinon ohjelmointiin ja ylläpitoon tarkoitettu, pääohjelmatasossa sekvenssikaavio-ohjelmoinnin ja aliohjelmissa graafisen lohkokaavio-ohjelmoinnin yhdistelmä. Robotino View:n päänäkymä muodostuu työkalurivistä ikkunan ylälaidassa, ohjelmanäkymästä ikkunan vasemmalla laidalla ja toimintalohko- ja muuttujalistasta ikkunan oikealla laidalla. [8,9]

Ohjelmistossa on ohjelmanprojektin luomisen lisäksi toiminnot yhteyden luomiseksi yhteen tai useampaan Robotinoon, Robotinon ohjelmointirajapinnan päivittämiseksi, firmwaren päivittämiseksi ja ohjelmaprojektin lataamiseksi Robotinon käyttöjärjestelmään. Erityisesti prototyyppin ohjelmointiin saadussa beta-vaiheen ohjelmistossa on valmiit toimilohkot myös kolmannen sukupolven Robotinon toimintoihin. [8]

Robotino View -ohjelmistossa on seuraavat opasrobotin toiminnassa käytettävät toiminnot:

- Robotin kinematiikkamalli robotin koordinaatiston suhteen
- Kiihtyvyysanturien ja sähkömoottorien antama robotin paikka- ja kulma-aproksimaatio ympäristön suhteen
- matriisikoodinlukija konenäölle
- Etäisyysmittaus laserskannerille
- Reitin ajo robotin kinematiikkamallin ja paikka-aproksimaation perusteella
- Analogiatulot infrapuna-antureille
- Binääritulo törmäyspuskurianturille
- AI, DI, DO -ohjelmointi muille liitettäville antureille ja toimilaitteille
- Kytkennät paikallisen PC:n tiedostojärjestelmään
- Kytkennät mahdollisen UDP-serverin luomiseen.

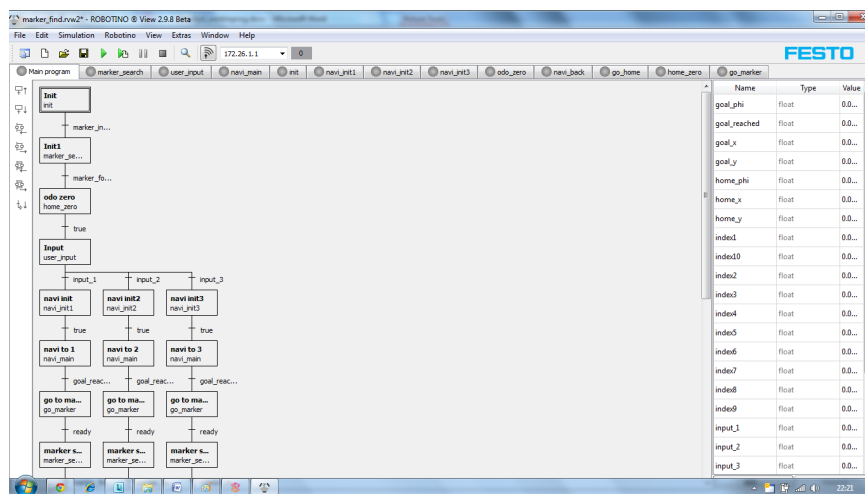
Robotino View:n päänäkö näkymä muodostuu työkalurivistä ikkunan ylälaidassa, ohjelmanäkymästä ikkunan vasemmalla laidalla ja toimintalohko- ja muuttujalistasta ikkunan oikealla laidalla. [8]

4.6.1 Robotino View -projekti

Projekti muodostuu pääohjelmasta (kuva 15) ja pääohjelman kutsumista aliohjelmista (kuva 16), sekä toimintalohkojen ja asetetuista parametreista. Robotino View -projekteja voidaan tallentaa ja ladata tietokoneen tiedostojärjestelmään rww-tiedostomuodossa. Liittessä 1 on esitelty prototyyppirobotin projekti.[8]

4.6.2 Pääohjelma

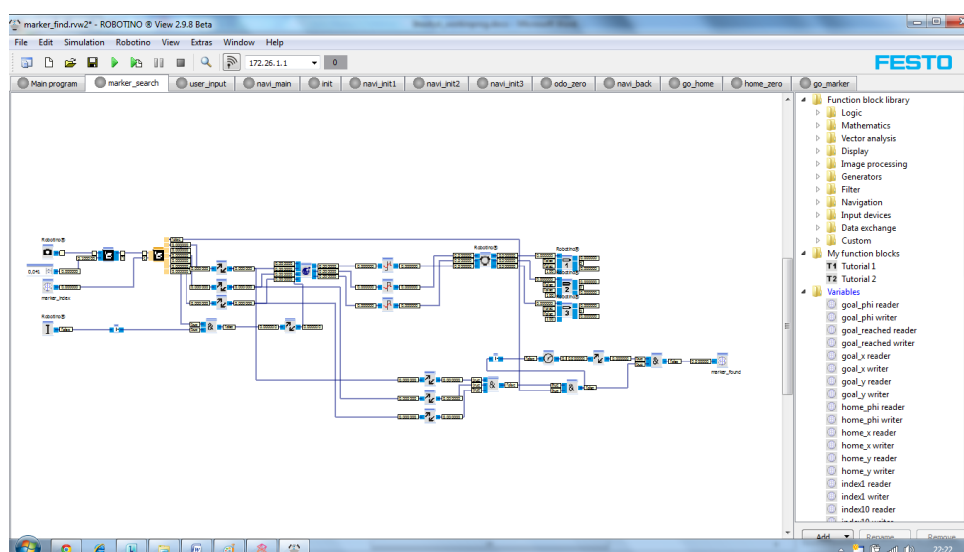
Robotino View -projektin pääohjelman suunnittelutyökalussa voidaan rakentaa sekvenssiohjelmia, jotka voidaan jakaa moniajoon tai ehtolausekkeiksi. Myös siirtyminen määriteltyyn ohjelmanosaan sekvenssin keskeltä on mahdollista. Etenemisehtoina toimivat globaalit liukulukumuuttujat, jotka määritellään ja joita ohjataan aliohjelmissa. Liukulukumuuttujat toimivat pääohjelmassa loogisen muuttujan tavoin, muuttujan ollessa nolla se on epätosi ja muuttujan ollessa jotain muuta se on tosi. Pääohjelman jokaiselle sekvenssilohkolle on annettava oma nimensä, mutta sekvenssilohkon suorittama aliohjelma voidaan valita. Tärkeintä on, että suoritettava aliohjelma vaikuttaa sekvenssiohjelman etenemisehtoon, jos lähtötilanne on epätosi. [8]



Kuva 15. Robotino View: Pääohjelman sekvenssiohjelmointinäkö

4.6.3 Aliohjelma

Aliohjelmat rakennetaan Robotino View:ssa graafisesti toimilohkoja liittämällä. Työnäkymän oikealla sijaitsevasta kirjastosta raahataan työtilaan halutut toimilohkot, ja toimilohkojen välille piirretään niiden kytkentäpisteiden muuttujat yhdistävät viivat. Aliohjelmiin voi lisäksi liittää itse tehtyjä API:in rakennettuja toimilohkoja ja Lua-ohjelmointikielellä rakennettuja komentosarjoja. Aliohjelmarakenteessa määritellään pääohjelmassa käytettävät liukulukumuuttujat. Aliohjelmissa etenemisehtomuuttujia voi käyttää normaalisti liukulukuina. Aliohjelmoinnissa on useita vain Robotino View:ssä käytettäviä muuttujamuotoja, joita voi käyttää myös pääohjelman globaali muuttujina [8].



Kuva 16. Robotino View: Aliohjelman lohko-ohjelmointinäkö

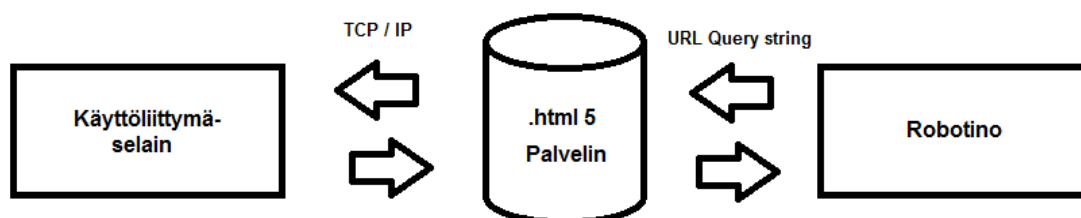
4.6.4 Muut ohjelmointitavat

Robotinoa voidaan ohjelmoida myös suoraan sulautettuun järjestelmään rakennetun avoimen lähdekoodin API:n kautta. Ohjelmointirajapinnat on määritelty muunmuassa C/C++, JAVA, .Net, LabVIEW, MATLAB/Simulink ja Microsoft Robotics Developer Studio -ohjelmointiympäristöihin. API:ssa on tuki kaikille laitteille, jotka on lisätty Robotino View -ympäristöön, mutta useimpiin alustoihin on saatavilla myös käyttäjien itse määrittelemiä ja jakamia muita laitteita ja ohjelmia. Käyttäjät voivat myös API:n tarjoaman tuen laajuudessa rakentaa omia toimilohkojaan Robotino View -ympäristöön. [8,9,10]

Tässä projektissa ei käytetä muita robotin ohjelmointitapoja, koska robotin toiminnan kannalta tärkeät toimilohkot ovat jo olemassa Robotino View -ohjelmassa. Niiden toiminnan kopioiminen itse tehtyyn ohjelmaan olisi erittäin työlästä, koska niiden lähdekoodi ei ole luettavassa muodossa. Oman ohjelman käyttö esimerkiksi API:n hyötyjen valjastamiseksi ei vähentäisi ohjaavan laitteen tarvetta käyttää myös Robotino View:tä robotin ohjaamiseen. Koska Robotino sallisi vain yhden ohjausyhteyden kerrallaan, täytyisi ohjaus rakentaa tässä sovelluksessa Robotino View:n kautta. [8]

4.7 Ohjelmistorakenne

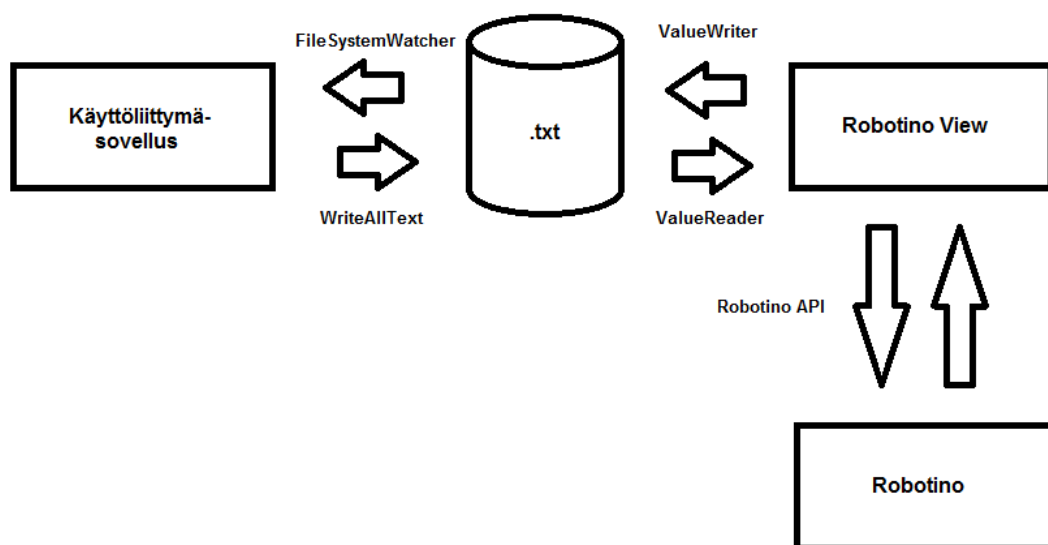
Lopullisessa opasrobotisovelluksessa Robotino View -ohjelmistolla tehtyt projektit ladataan suoritettaviksi robotin reaaliaikakäyttöjärjestelmään, jossa on sisäänrakennettuna oma paikallisesti Robotino View -projekteja suorittava ohjelmarakenteensa. Projektit kutsutaan suoritettaviksi IP-liikenteellä URL:n kyselymerkkijonoon liitettyllä käynnistyskomennolla, joka lähetetään Robotinossa olevalta palvelimelta. Käyttöliittymäsivu, joka sijaitsee palvelimella rakennetaan HTML 5 -muodossa. Käyttöliittymälaitte voi näin olla mikä tahansa HTML 5 yhteensopivan selaimen sisältävä laite. Kuvassa 17 on esitelty järjestelmän rakenne. [10]



Kuva 17. Opasrobotin ohjauksen rakenne

Prototyypissä projekteja käytetään Robotino View:llä Robotinon kanssa lähiverkkoon kytketyllä tietokoneella. Käyttäjärajapinta prototyypissä on Visual Studio 2013:lla luotu GUI, joka käyttää tiedonvaihtoon Robotino View:n kanssa FileSystemWatcher-nimistä .NET Framework 4.5 System.IO -luokan tapahtumaa ja tiedostoonkirjoituskomentoja sekä ohjaavassa sovelluksessa että Robotino View:ssä. Robotino View kirjoittaa muuttujansa tiedostoon ja käyttöliittymäsovellus tutkii, milloin mitään muuttujaa

kirjoitetaan. Kirjoittamalla tiedostojärjestelmään ristiin kukin vuorollaan ohjelmat luovat rajapinnan, jolla välitetään robotin tila- ja paikkatietoja robotin ohjauksesta käyttöliittymään ja päinvastoin. Tämä rajapinta on helppo toteuttaa, ja siihen on nopeaa tehdä pieniä muutoksia tarvittaessa. Robotino View:ssä on myös mahdollisuus rakentaa UDP-serveri, mutta projektin aikaresurssit eivät antaneet mahdollisuutta tutustua tähän toiminnallisuuteen tarkemmin. Kuvassa 18 on esitelty järjestelmän rakenne. [8,14,15]



Kuva 18. Opasrobotin prototyypin ohjauksen rakenne

5 Kehitysprojekti

Robottijärjestelmän kehityksen alkuvaiheessa syksyllä 2012 ja keväällä 2013 määriteltiin sovelluksen käyttökohteen olosuhteet, käytettävä kehitysalusta, toiminnot sekä projektin aikataulu. Projektin osapuolten välillä esiteltiin ideoita, eriteltiin kohteen ominaisuuksia, arvioitiin toteutusmahdollisuuksia ja sovittiin vähimmäisvaatimuksista.

Tässä kappaleessa eritellään kevään, kesän ja syksyn 2013 aikana tehdyn tutkimustyön tulosten perusteella määritellyn projektin yksityiskohtia.

5.1 Projektin kuvaus

Robottijärjestelmä muodostuu vapaasti näyttelytilassa liikkuvista kahdesta robotista ja niiden tukikohdasta. Robotin avulla on tarkoitus kertoa yleistajuisesti sekä vapaasti liikkuviin robotteihin liittyvistä teknisistä haasteista että robotteihin liittyvien tekoälyratkaisujen kehityksestä erityisesti kommunikoinnissa ihmisten kanssa. Robottijärjestelmä toteutetaan niin, että sekä sen ylläpito että edelleen kehittäminen on teknisesti mahdollista. Robottijärjestelmän elinkaaren arvoidaan olevan kokonaisuudessaan noin viisi vuotta. Robottijärjestelmä sijoitetaan alustavasti Heureka sylinterihalliin, johon tulee myös sen tukikohta eli latauspiste.

Kehitystyön ensimmäinen vaihe sijoittuu ajalle lokakuusta 2013 helmikuuhun 2014. Ensimmäisessä vaiheessa rakennetaan robotit fyysisesti valmiiksi automaattisia latausasemia ja sen toimintoja lukuun ottamatta, suunnitellaan käyttäjärajapinnan perusrakenne, ja tehdään robotin navigointijärjestelmä valmiiksi. Ensimmäisessä työvaiheessa tehty perusrakenne järjestelmälle mahdollistaa toisen vaiheen jatkokehityksen. Vaiheen toteuttaa Festo Didactic.

Kehitystyön toinen vaihe sijoittuu ajalle helmikuusta 2014 kesäkuuhun 2014. Tässä vaiheessa kehitetään automaattinen latausjärjestelmä, luodaan senhetkisten mahdollisuuksien rajoissa mediasisältöä käyttäjärajapintaan sekä suoritetaan toiminnan viimeistely sopivaksi toimintaympäristöön. Toisen vaiheen toteutuksesta vastaa latausjärjestelmän kehitystyötä ja käyttöönottoimenpiteitä lukuun ottamatta Heureka. Festo Didactic antaa tarvittaessa teknistä tukea mediasisältöjen upottamiseksi käyttäjärajapintaan, kustannusten jaosta sovitaan etukäteen.

Käyttövaiheessa Heureka arvioi ja raportoi vapaamuotoisesti Festolle robottijärjestelmän käyttöä ja toimivuutta. Festo Didactic tarjoaa tarvittaessa teknistä tukea ja jatkokehitystyötä.

5.2 Liikkuminen ja toiminta

Robotti liikkuu vapaasti etukäteen määritellyllä alueella törmäämättä esineisiin tai ihmisiin tai muihin asioihin. Tarkka toiminta-alue määritellään tarkemmin myöhemmin jatkosuunnittelun yhteydessä. Liikkuminen perustuu opetettuun tukipisteen ja

maalipisteiden sekä niiden välisten reittipisteiden opettamalla muodostettuun koordinaattipistejärjestelmään. Törmäyksen estoon ja reitinhakuun käytetään laserskanneria, infrapuna-antureita ja poikkeustilanteissa törmäyspuskuria.

5.3 Toiminta-aika

Robottien tulevat palvelemaan yleisöä Heurekan normaaleina aukioloaikoina. Tämä tarkoittaa robottijärjestelmälle jatkuvia 7 – 10 tunnin työjaksoja. Heureka on auki viikonpäivistä ja vuodenaajoista riippuen yleensä kello 10 – 17, 10 – 18 tai 10 – 20. On arvioitu, että kaksi robottia varustettuna 24 voltin 18 ampeeritunnin akustolla suoriutuu tehtävästä vuoroissa. Prototyypissä on 24 voltin 8 ampeeritunnin akusto, jonka teho riittää noin kahden ja puolen tunnin käyttöön. [10]

5.3.1 Automaattinen latauspiste

Automaattinen lataustoiminto otetaan mukaan kehitystyöhön toisessa vaiheessa. Tavoitteena kehitystyössä on, että robotin tulee kyetä hakeutumaan ja telakoitumaan latauspistokkeeseensa automaattisesti ennen kuin sen akuista loppuu virta. Toinen tavoite on, että robotin tulee osata lähteä automaattisesti jälleen liikkeelle, kun sillä on akuissaan riittävä lataus. Toimiakseen kahden robotin välillä saumattomasti siten, että yksi robotti on aina vuorossa, automaattinen latausjärjestelmä vaatii myös robottien välistä kommunikaatiota. Tämä vaatii robottien verkkoympäristön laajennusta.

Ensimmäisessä vaiheessa kehitetään toiminnallisuus, jossa robotti hakeutuu latauspisteeseen sen akkujen latauksen ollessa vähäinen. Robotti sammuttaa itsensä, ja jää odottamaan, että se laitetaan lataukseen. Robottien toiminta-ajan arvioidaan olevan niin pitkä, ettei automaattista latauspistettä tarvita työpäivän aikana muuhun kuin käyttöhenkilökunnan käyttömukavuuden lisäämiseen. Ensimmäisen vaiheen toiminta vaatii, että työpäivän alussa robotti laitetaan päälle manuaalisesti painonapilla, ja kytketään lataukseen robotin työjakson lopussa. Toisen vaiheen kuluessa päätetään, kumpi toiminnallisuus valitaan käyttövaiheeseen.

5.4 Paikannus

Ensimmäisessä vaiheessa toteutetaan paikannus, joka perustuu robotin gyroskoopin ja kiihtyvyyssanturoinnin sekä moottorien inkrementtianturien laskennan antamaan likiarvoiseen koordinaattiin, sekä toiminta-alueella oleviin referenssimatriisikoodeihin, jotka luetaan konenäöllä. Prototyypillä suoritettujen koeajojen perusteella tällä mittauksella arvioidaan päästävän riittävään toimintatarkkuuteen (kuva 19). [10]



Kuva 19. Opasrobotin prototyyppi tunnistamassa matriisikoodin

5.5 Käyttäjäraja

Robotin rajapinta perustuu sulautetun järjestelmän palvelimella olevaan HTML-sivuun, jota selataan joko paikallisesti kosketusnäytöllä tai robotin lähiverkon kautta tabletilla. Kehitystyön ensivaiheessa luodaan jatkokehitykselle avoin käyttäjäraja, jolla voidaan hallinnoida robottiohjelmaa sekä ottaa vastaan robotin tilatietoja esimerkiksi navigoinnista. Näillä toteutetaan kohteen perustoiminta. [10]

Toisessa vaiheessa, kun perusjärjestelmä on saatu toimivaksi, voidaan kehittää Heureka HTML-kehitysresurssien puitteissa käyttäjärajan mediaa ja interaktiivisuutta monipuolisemmaksi. Jatkokehityksellä verkkoympäristössä voidaan myös lisätä robottien vuorovaikutusta, esimerkiksi automaattista käytön ohjautumista.

5.6 Turvallisuus

Robotti on autonomisesti ohjautuva, mutta sitä voidaan ohjata myös manuaalisesti langattomalla ohjaimella. Robotissa on kolme moottoria, jotka liikuttavat robotin runkoa holonomisesti. Navigoinnin ja robotin liikkumisen arvioidaan olevan turvallista, koska ohjelma rakennetaan siten, että vain varma mittaustieto aiottuun liikesuuntaan antaa nopeusohjeen moottorille. Ohjelma tehdään siten, että mikäli mikä tahansa anturi vikaantuu, estää se robotin toiminnan ilman anturin tietoa. Mikäli jokin moottoreista vikaantuu tai ilmaantuu jokin muu mekaaninen vika, joka aiheuttaa sen, ettei robotti liiku nopeusohjeen määrittämään suuntaan, tehdään ohjelmallisesti virheilmoitus ja robotti keskeyttää toimintansa. Kaikki käytettävä sähkölaiteisto toimii suojajännitealueella, joten laitteiston käyttö on turvallista myös maallikoille. [9,10]

5.6.1 Mekaaniset vaarat

Mahdollisessa törmäyksessä tai puristumis/kiilautumistilanteessa robotin massa on aikuisen ihmisen voimin helppo siirtää pois tieltä, vaikka moottorit ohjaisivat täydellä teholla siirtosuuntaan. Ohjelma rakennetaan siten, että robotin suurin mahdollinen nopeus on rauhallinen kävelyvauhti. Robotin suurin mahdollinen kiihtyvyys on turvallisella tasolla johtuen moottorien rajallisesta tehosta. Runko ja kuoren rakenne suunnitellaan siten, ettei siinä ole puristuksiin tai jumiin jäämisen mahdollisuutta aiheuttavia onkaloita tai reikiä, teräviä viiltohaavoja tai muita lieviä vammoja mahdollisesti aiheuttavia reunoja, tai robotin mahdollisessa pystykeskiakselin suuntaisessa pyörimisliikkeessä nopeasti liikkuvia vipuvarsia. Pienten lasten tulee käyttää robottia aikuisen valvonnassa. Työjakson aikainen käyttö on ennalta määritellyn vastuullisen käyttöhenkilökunnan valvomaa. [9]

5.6.2 Varolaitteet

Robottiin asennetaan ensimmäisessä vaiheessa hätäseis-kytkin joka kytkee hätätilanteessa robotista päävirran pois. Tällöin robotin moottoriohjaimet ja keskusyksikkö sammuvat. Kuittaus tapahtuu käynnistämällä robotti uudelleen. Kehitystyön aikana tulee arvioida, aiheuttavatko esimerkiksi lapset robotin tahattomia poiskytkentöjä painamalla näkyvillä olevaa punaista nappia.

5.7 Toiminta

Ensimmäisessä vaiheessa rakennetun kokoonpanon toiminta:

- Robotti odottaa tukipisteessään käyttäjän valintaa (kielen valinta, toiminnon valinta)
- Valinnan saatuaan robotti navigoi valittuun päämäärään
- Robotti odottaa päämäärässä käyttäjän kuittausta / aikakatkaisua
- Robotti navigoi takaisin tukipisteeseen.

Toisessa vaiheessa rakennetun kokoonpanon toiminta:

- Robotti odottaa tukipisteessään käyttäjän valintaa, soittaa mediaa aktiivisesti herättääkseen huomiota
- Käyttäjän valintojen perusteella soittaa mediaa / interaktiiviset toiminnot
- Päämäärävalinnan saatuaan navigoi päämäärään, median soitto (kertoo kohteesta yms.)
- Päämäärässä mahdolliset interaktiiviset toiminnot käyttäjän kanssa, kuittaus palata
- Navigointi takaisin tukipisteeseen
- Uusi valinta / lataukseen ajo / latauksesta ajo.

6 Yhteenveto

Alkuvaiheen työ käsitti paljon robottikonseptin hahmottelua. Vision opasrobotista muotoutuessa vähitellen todelliseksi suunnitelmaksi robottijärjestelmän kehittämiseksi, jäi paljon robotin ensimmäisissä määrittelyissä mainittuja toiminnallisuuksia pois lopullisesta toimituksesta. Projektin luonnetta muokattiin vahvasti varsinkin sen alkuvaiheessa johtuen pääosin kehitysresurssien rajallisuudesta. Tieteisfiktio ja toisaalta alan huippututkimuksen esittelemien saavutusten aiheuttamat ennakkokäsitykset robotiikan toteutusmahdollisuuksista saivat aliarvioimaan tarvittavan työn määrää. Kehitysalustan ja ohjelmarungon valinnassa pyrittiin perustoimintojen tuottamisen ja helpon rakentamisen lisäksi siihen, että toiminnallisuuksien lisääminen tulevaisuudessa olisi mahdollista.

Uuden Robotino-sukupolven esittely syksyllä 2013 aiheutti pienen viivästyksen lopulliseen projektin toteutukseen, koska alun perin projekti piti tehdä toisen sukupolven Robotinolla. Prototyypin muokkaus uuden sukupolven toimintoja käyttäväksi järjestelmäksi onnistui hyvin Feston Saksan pääkonttorin antaman teknisen tuen avulla.

Robotin prototyyppi jää alustavasti Metropolia Ammattikorkeakoululle opetus- ja tutkimuskäyttöön. Sillä voidaan mahdollisesti tuottaa tulevaisuudessa lisää samankaltaista sovelluskohteiden valmistelutyötä. Itse Heurekan robottiprojekti odottaa vielä toteuttamistaan tämän insinööriyön kirjoitushetkellä. Valmistelut on tehty hyvin pitkälle, mutta lopullinen toteutus on tekemättä. Prototyypin avulla tehtyjen kokeilujen perusteella tulee Heurekassa olemaan toimiva opasrobotijärjestelmä kesäkuusta 2014 alkaen.

Opasrobotin toimittaminen on laaja projekti, joka vaatii paljon kehitystyötä myös käyttökohteessaan. Siitä on tehty suhteellisen paljon teorian tasoa tutkimusta. Robotiikka on vasta kehittymässä oleva ala, joten oikeasti toiminnassa olevat järjestelmät ovat harvassa. Toivottavasti tämä työ edesauttaa yleisesti korkean tason teknologiana pidetyn robotiikan arkipäiväistymistä.

Lähteet

Yhteistyösopimus 28.10.2013. Festo Oy ja Heureka

Suullisesti sovitut toimintatavat Heurekassa tammikuu 2013 – lokakuu 2013. Festo Oy ja Heureka

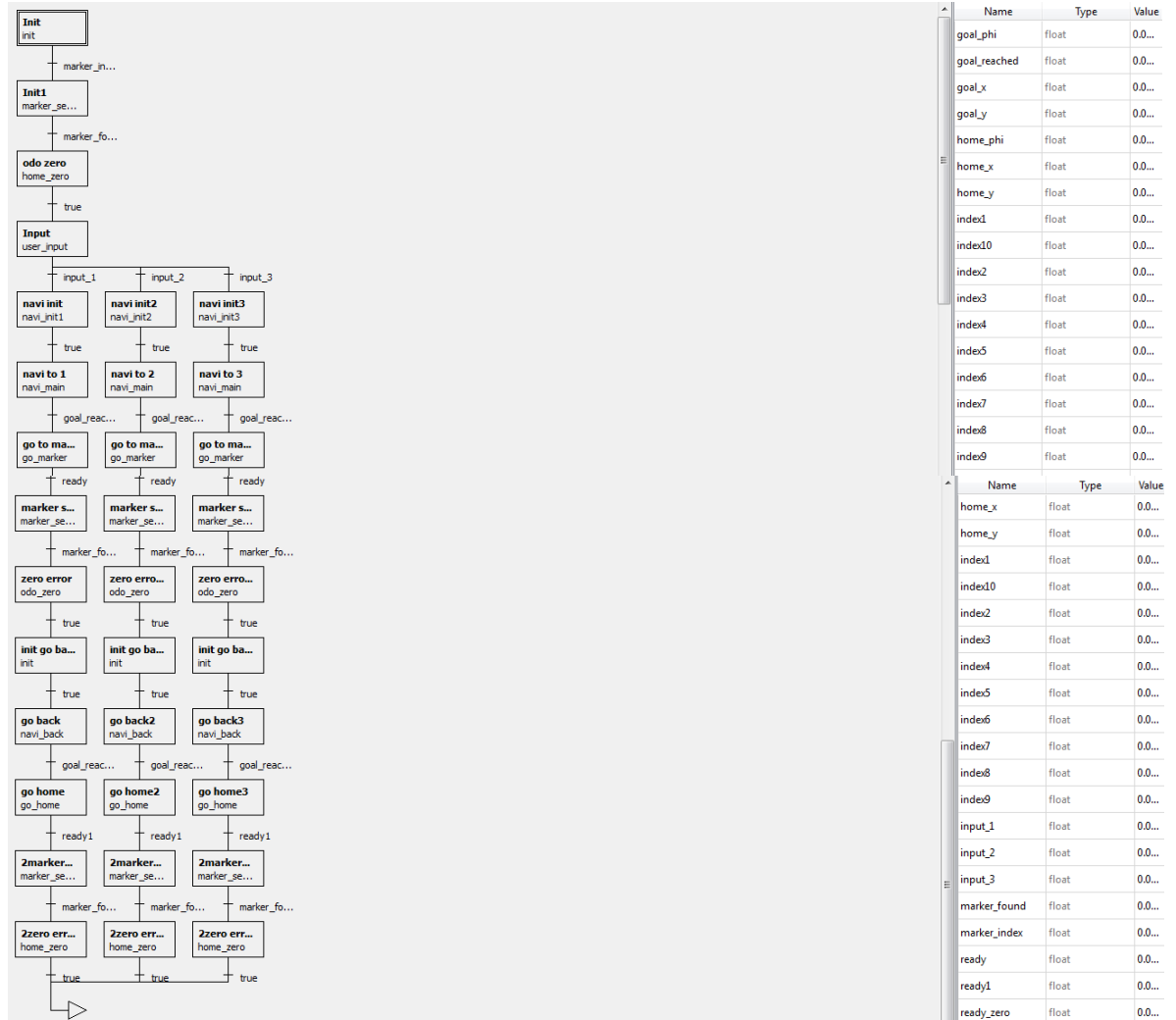
Keskustelut robotiikan tohtori Dirk Penskyn kanssa. tammikuu-lokakuu 2013. Festo Didactic GmbH & Co. KG, 73770 Denkendorf, Saksa.

- 1 Gates, B. 2007. A robot in Every Home. Artikkel Scientific American -lehdessä, Yhdysvallat.
- 2 Kracht, S. – Nielsen, C. 2007. Robots in Everyday Human Environments. Aalborg University, Tanska.
- 3 Dedola, F. 2007. Quality in Human-Robot Encounters. Aalborg University, Tanska.
- 4 Bina, D. 2010 A study on local planning techniques for mobile robot navigation. University of Coimbra, Portugali
- 5 Brandao, A.S, Sarcinelli-Filho, M. – Carelli, R. 2013. An Analytical Approach to Avoid Obstacles in Mobile Robot Navigation. International Journal of Advanced Robotic Systems. Intect.
- 6 Rodriguez-Losada, D. – Matia, F – Galan, R. – Hernando, M. – Montero, J.M. – Lucas, J.M. 2008. Urbano, an Interactive Mobile Tour-Guide Robot. Universidad Politecnica de Madrid, Espanja.
- 7 Ferreira, A. - Pereira, F.G. - Vassallo, R.F - Bastos, T.F. - Sarcinelli, M. 2008. An approach to avoid obstacles in mobile robot navigation: the Tangential Escape. Universidade Federal do Espírito Santo, Brasilia.
- 8 Robotino® View2.9.8Beta Help Contents. 2013. Festo Didactic GmbH & Co. KG, 73770 Denkendorf, Saksa.
- 9 Robotino Manual. 2010. Festo Didactic GmbH & Co. KG, 73770 Denkendorf, Saksa
- 10 Technology for education and science: The current range of Festo Didactic products 2014. 2013. Festo Didactic GmbH & Co. KG, 73770 Denkendorf, Saksa

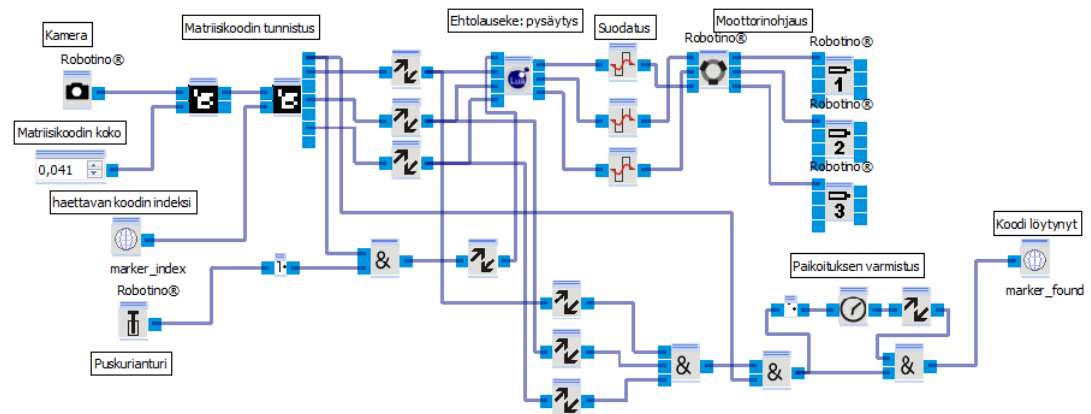
- 11 Thrun, S. – Bennewitz, M. – Burgard, W. – Cremers, A.B. – Dellaert, F. – Fox, D. – Hähnel, D. – Lakemeyer, G. – Rosenberg, C. – Roy, N. – Schulte, J. – Schulz, D. – Steiner, W.. 1999. Experiences with two Deployed Interactive Tour-Guide Robots. School of Computer Science Carnegie Mellon University, Yhdysvallat. Computer Science department III University of Bonn, Saksa. Computer Science Department Aachen University of Technology, Saksa.
- 12 Thrun, S. – Bennewitz, M. – Burgard, W. – Cremers, A.B. – Dellaert, F. – Fox, D. – Hähnel, D. – Lakemeyer, G. – Rosenberg, C. – Roy, N. – Schulte, J. – Schulz, D. – Steiner, W.. 1998. The InteractiveMuseum Tour-Guide Robot. School of Computer Science Carnegie Mellon University, Yhdysvallat. Computer Science department III University of Bonn, Saksa. Computer Science Department Aachen University of Technology, Saksa.
- 13 Thrun, S. – Bennewitz, M. – Burgard, W. – Cremers, A.B. – Dellaert, F. – Fox, D. – Hähnel, D. – Rosenberg, C. – Roy, N. – Schulte, J. – Schulz, D. – Steiner, W.. 1999. MINERVA: A Second-Generation Museum Tour-Guide Robot. School of Computer Science Carnegie Mellon University, Yhdysvallat. Computer Science department III University of Bonn, Saksa.
- 14 FileSystemWatcher. Verkkodokumentti. Microsoft. [http://msdn.microsoft.com/en-us/library/system.io.filesystemwatcher\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.io.filesystemwatcher(v=vs.110).aspx). Luettu 18.10.2013
- 15 WriteAllText. Verkkodokumentti. Microsoft. [http://msdn.microsoft.com/en-us/library/system.io.file.writealltext\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.io.file.writealltext(v=vs.110).aspx). Luettu 18.10.2013
- 16 Laser rangefinder. Verkkodokumentti.Wikipedia. http://en.wikipedia.org/wiki/Laser_rangefinder. luettu 12.9.2013
- 17 TPR Robina. Verkkodokumentti. Plasticpals. <http://www.plasticpals.com/?p=16011>. luettu 13.9.2013
- 18 Festo yritystietoa. Verkkodokumentti. Festo. http://www.festo.com/cms/fi_fi/index.htm. luettu 10.9.2013
- 19 Festo yritystietoa. Verkkodokumentti. Wikipedia. <http://en.wikipedia.org/wiki/Festo>. luettu 10.9.2013
- 20 Festo Didactic yritystietoa. verkkodokumentti. Festo Didactic. <http://www.festo-didactic.com/int-en/>. luettu 10.9.2013
- 21 Tiedekeskus Heureka. Verkkodokumentti. Wikipedia, http://fi.wikipedia.org/wiki/Tiedekeskus_Heureka. luettu 10.9.2013
- 22 Robotino kinematiikka. verkkodokumentti. Aalborg University. <http://www.control.aau.dk/~tb/wiki/index.php/Kinematics>. luettu 5.10.2013

Opasrobotin prototyypin navigointiohjelmien pääosat

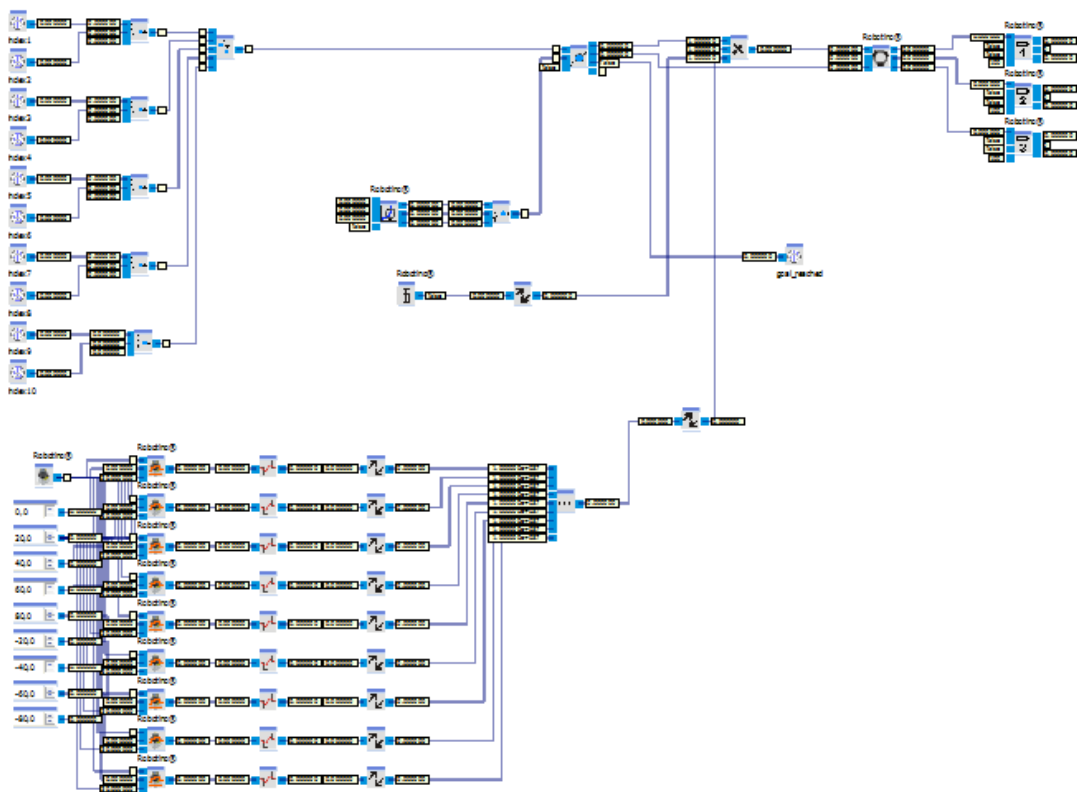
Tässä liitessä eritellään tärkeimmät navigointiin liittyvät ohjelmaosiot



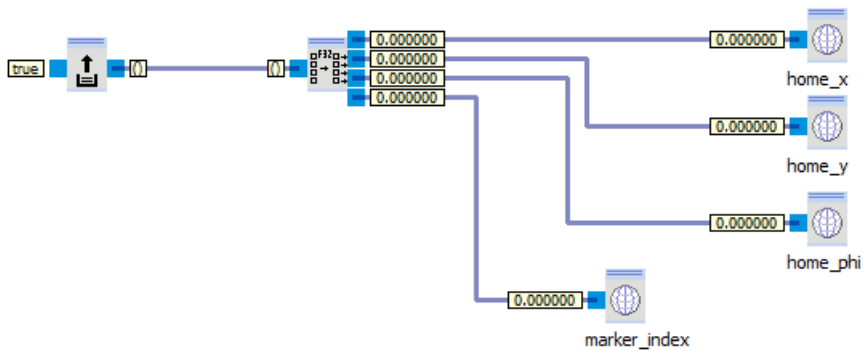
Kuva 20. Pääohjelma: Navigointi kolmen opetetun pisteen ja kotipisteen välillä



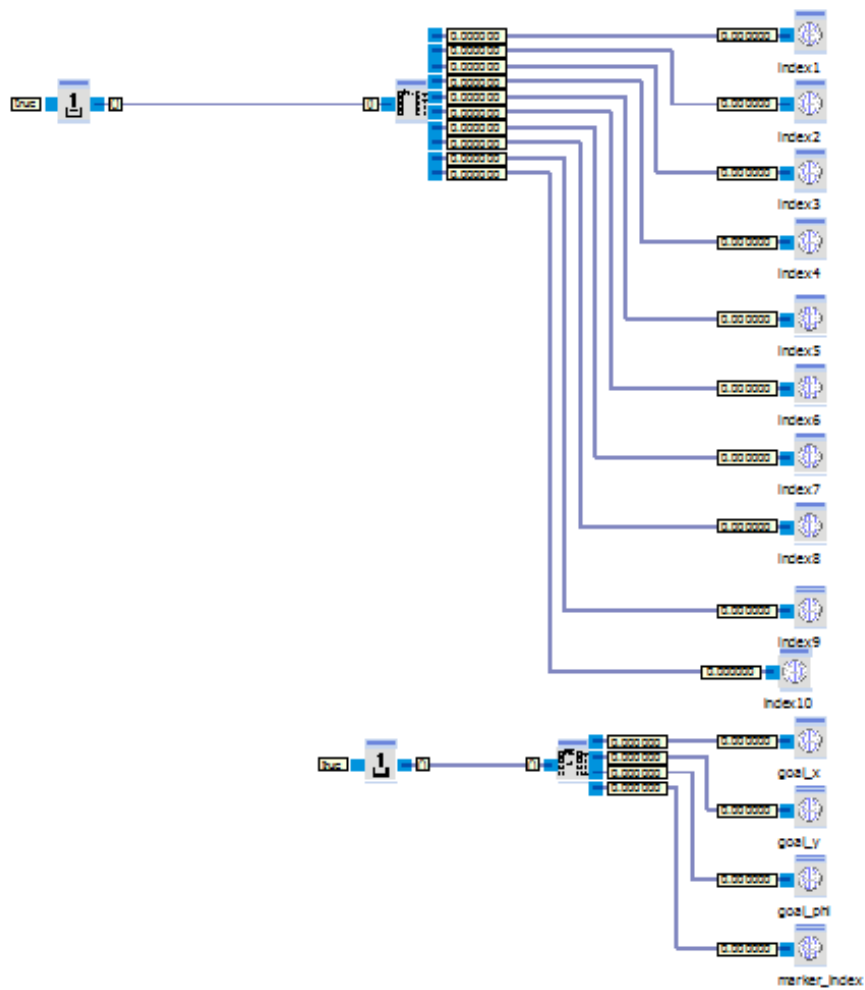
Kuva 21. Aliohjelma: Määrätyn matriiskoodin etsintä ja robotin linjaus koodin kanssa



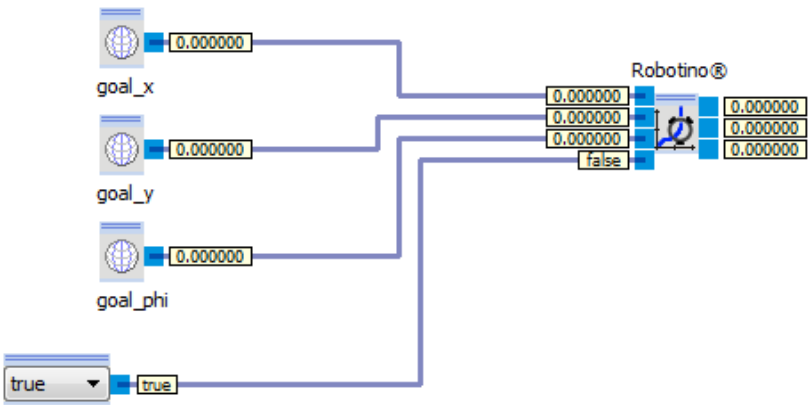
Kuva 22. Aliohjelma: Navigointi reittitietokannasta luetun reitin perusteella



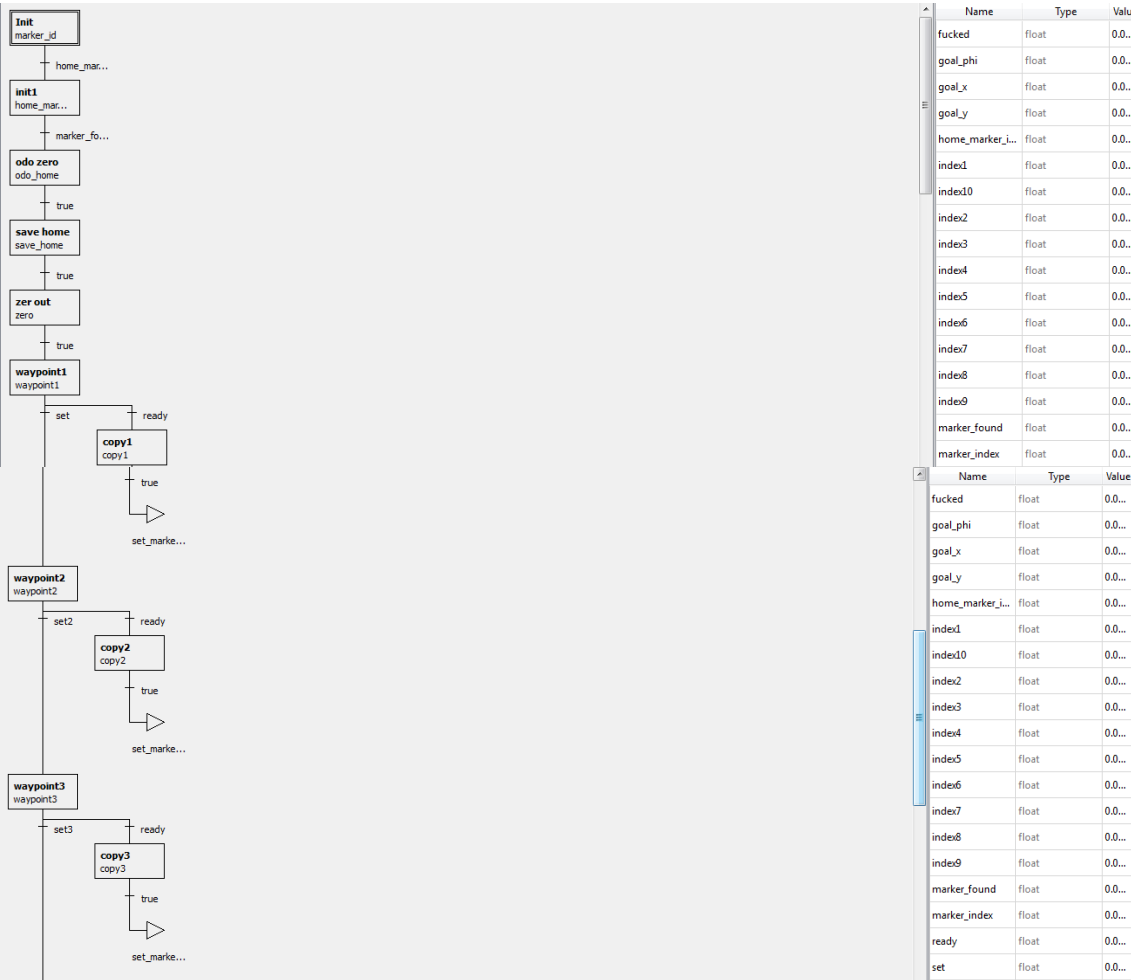
Kuva 23. Aliohjelma: Kotiaseman koordinaattivektorin haku tiedostosta ja purku muuttujiksi

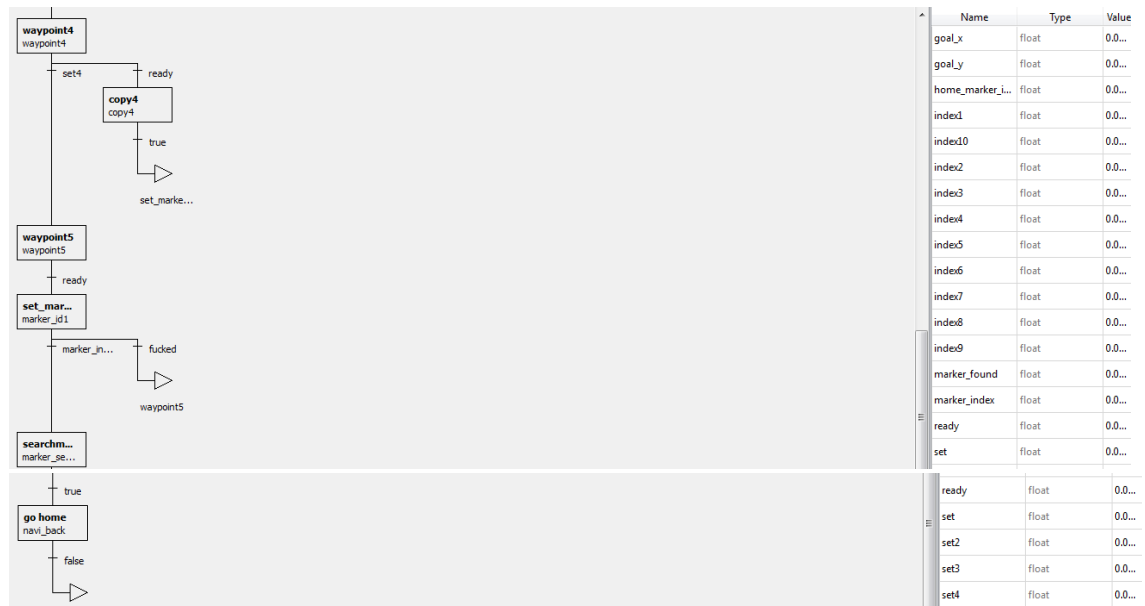


Kuva 24. Aliohjelma: Yhden maalipisteen reitin ja koordinaattien haku tiedostosta ja purku pääohjelman muuttujiksi

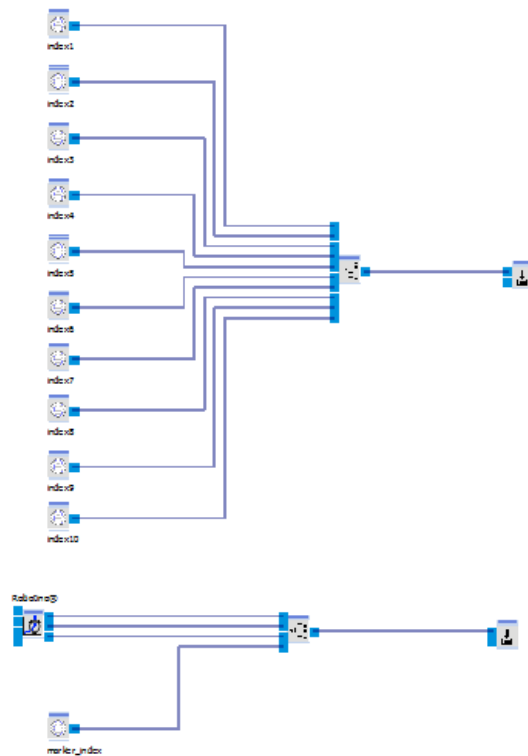


Kuva 25. Aliohjelma: Maalin löytämisen jälkeen aiheutuneen virheen nollaus paikkatietoon





Kuva 26. Pääohjelma: Manuaalinen reitin ja maalipisteen opetus



Kuva 27. Reittitiedon ja matriisikoodin löytämisen jälkeen luetun paikkatiedon kirjoitus tietokantaan.

